CHAPTER

6

POWER FLOW ANALYSIS

6.1 INTRODUCTION

In the previous chapters, modeling of the major components of an electric power system was discussed. This chapter deals with the steady-state analysis of an interconnected power system during normal operation. The system is assumed to be operating under balanced condition and is represented by a single-phase network. The network contains hundreds of nodes and branches with impedances specified in per unit on a common MVA base.

Network equations can be formulated systematically in a variety of forms. However, the node-voltage method, which is the most suitable form for many power system analyses, is commonly used. The formulation of the network equations in the nodal admittance form results in complex linear simultaneous algebraic equations in terms of node currents. When node currents are specified, the set of linear equations can be solved for the node voltages. However, in a power system, powers are known rather than currents. Thus, the resulting equations in terms of power, known as the *power flow equation*, become nonlinear and must be solved by iterative techniques. Power flow studies, commonly referred to as *load flow*, are the backbone of power system analysis and design. They are necessary for planning, operation, economic scheduling and exchange of power between utilities. In addition, power flow analysis is required for many other analyses such as transient stability and contingency studies.

In this chapter, the bus admittance matrix of the node-voltage equation is formulated, and a *MATLAB* function named **ybus** is developed for the systematic formation of the bus admittance matrix. Next, two commonly used iterative techniques, namely Gauss-Seidel and Newton-Raphson methods for the solution of nonlinear algebraic equations, are discussed. These techniques are employed in the solution of power flow problems. Three programs **lfgauss, lfnewton,** and **decouple** are developed for the solution of power flow problems by Gauss-Seidel, Newton-Raphson, and the fast decoupled power flow, respectively.

6.2 BUS ADMITTANCE MATRIX

In order to obtain the node-voltage equations, consider the simple power system shown in Figure 6.1 where impedances are expressed in per unit on a common MVA base and for simplicity resistances are neglected. Since the nodal solution is based upon Kirchhoff's current law, impedances are converted to admittance, i.e.,

$$y_{ij} = \frac{1}{z_{ij}} = \frac{1}{r_{ij} + jx_{ij}}$$

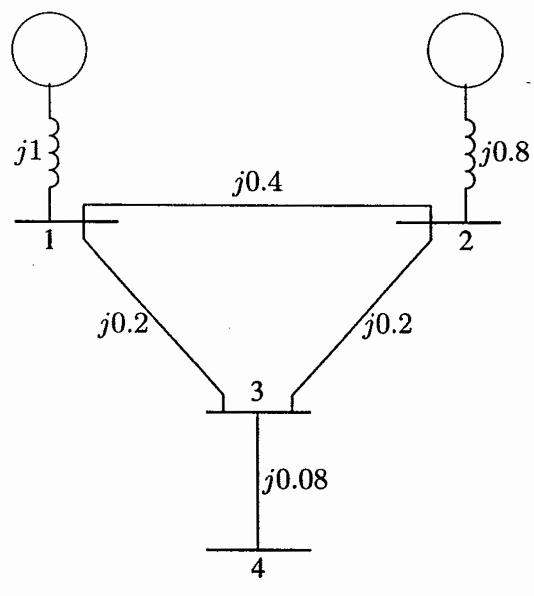

FIGURE 6.1
The impedance diagram of a simple system.

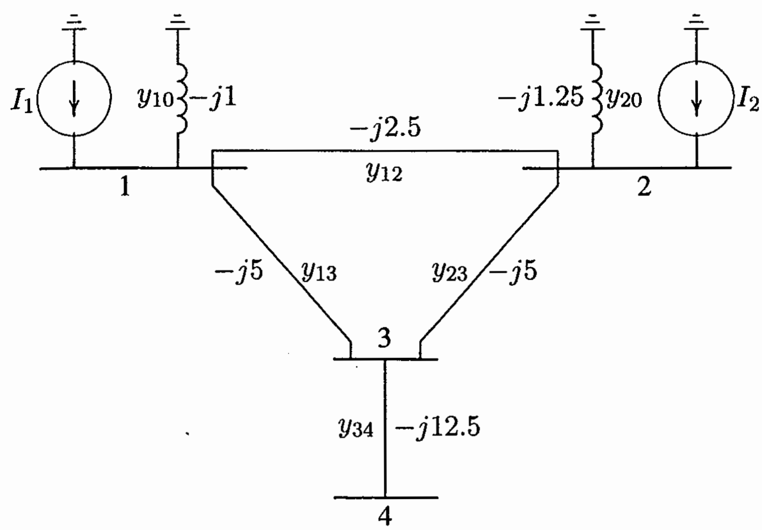

FIGURE 6.2
The admittance diagram for system of Figure 6.1.

The circuit has been redrawn in Figure 6.2 in terms of admittances and transformation to current sources. Node 0 (which is normally ground) is taken as reference. Applying KCL to the independent nodes 1 through 4 results in

$$\begin{aligned}
I_1 &= y_{10}V_1 + y_{12}(V_1 - V_2) + y_{13}(V_1 - V_3)\\
I_2 &= y_{20}V_2 + y_{12}(V_2 - V_1) + y_{23}(V_2 - V_3)\\
0 &= y_{23}(V_3 - V_2) + y_{13}(V_3 - V_1) + y_{34}(V_3 - V_4)\\
0 &= y_{34}(V_4 - V_3)
\end{aligned}$$

Rearranging these equations yields

$$\begin{aligned}
I_1 &= (y_{10} + y_{12} + y_{13})V_1 - y_{12}V_2 - y_{13}V_3\\
I_2 &= -y_{12}V_1 + (y_{20} + y_{12} + y_{23})V_2 - y_{23}V_3\\
0 &= -y_{13}V_1 - y_{23}V_2 + (y_{13} + y_{23} + y_{34})V_3 - y_{34}V_4\\
0 &= -y_{34}V_3 + y_{34}V_4
\end{aligned}$$

We introduce the following admittances

$$\begin{aligned}
Y_{11} &= y_{10} + y_{12} + y_{13}\\
Y_{22} &= y_{20} + y_{12} + y_{23}
\end{aligned}$$

$$
\begin{aligned}
Y_{33} &= y_{13} + y_{23} + y_{34} \\
Y_{44} &= y_{34} \\
Y_{12} &= Y_{21} = -y_{12} \\
Y_{13} &= Y_{31} = -y_{13} \\
Y_{23} &= Y_{32} = -y_{23} \\
Y_{34} &= Y_{43} = -y_{34}
\end{aligned}
$$

The node equation reduces to

$$
\begin{aligned}
I_1 &= Y_{11}V_1 + Y_{12}V_2 + Y_{13}V_3 + Y_{14}V_4 \\
I_2 &= Y_{21}V_1 + Y_{22}V_2 + Y_{23}V_3 + Y_{24}V_4 \\
I_3 &= Y_{31}V_1 + Y_{32}V_2 + Y_{33}V_3 + Y_{34}V_4 \\
I_4 &= Y_{41}V_1 + Y_{42}V_2 + Y_{43}V_3 + Y_{44}V_4
\end{aligned}
$$

In the above network, since there is no connection between bus 1 and 4, $Y_{14} = Y_{41} = 0$; similarly $Y_{24} = Y_{42} = 0$.

Extending the above relation to an n bus system, the node-voltage equation in matrix form is

$$
\begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_i \\ \vdots \\ I_n \end{bmatrix} = \begin{bmatrix} Y_{11} & Y_{12} & \cdots & Y_{1i} & \cdots & Y_{1n} \\ Y_{21} & Y_{22} & \cdots & Y_{2i} & \cdots & Y_{2n} \\ \vdots & \vdots & & \vdots & & \vdots \\ Y_{i1} & Y_{i2} & \cdots & Y_{ii} & \cdots & Y_{in} \\ \vdots & \vdots & & \vdots & & \vdots \\ Y_{n1} & Y_{n2} & \cdots & Y_{ni} & \cdots & Y_{nn} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_i \\ \vdots \\ V_n \end{bmatrix} \tag{6.1}
$$

or

$$
\mathbf{I}_{bus} = \mathbf{Y}_{bus} \, \mathbf{V}_{bus} \tag{6.2}
$$

where $\mathbf{I}_{bus}$ is the vector of the injected bus currents (i.e., external current sources). The current is positive when flowing towards the bus, and it is negative if flowing away from the bus. $\mathbf{V}_{bus}$ is the vector of bus voltages measured from the reference node (i.e., node voltages). $\mathbf{Y}_{bus}$ is known as the *bus admittance matrix.* The diagonal element of each node is the sum of admittances connected to it. It is known as the *self-admittance* or *driving point admittance*, i.e.,

$$
Y_{ii} = \sum_{j=0}^{n} y_{ij} \qquad j \neq i \tag{6.3}
$$

The off-diagonal element is equal to the negative of the admittance between the nodes. It is known as the *mutual admittance* or *transfer admittance*, i.e.,

$$
Y_{ij} = Y_{ji} = -y_{ij} \tag{6.4}
$$

When the bus currents are known, (6.2) can be solved for the n bus voltages.

$$\mathbf{V}_{bus} = \mathbf{Y}_{bus}^{-1}\,\mathbf{I}_{bus} \tag{6.5}$$

The inverse of the bus admittance matrix is known as the *bus impedance matrix* Z_{bus}. The admittance matrix obtained with one of the buses as reference is nonsingular. Otherwise the nodal matrix is singular.

Inspection of the bus admittance matrix reveals that the matrix is symmetric along the leading diagonal, and we need to store the upper triangular nodal admittance matrix only. In a typical power system network, each bus is connected to only a few nearby buses. Consequently, many off-diagonal elements are zero. Such a matrix is called *sparse*, and efficient numerical techniques can be applied to compute its inverse. By means of an appropriately ordered triangular decomposition, the inverse of a sparse matrix can be expressed as a product of sparse matrix factors, thereby giving an advantage in computational speed, storage and reduction of round-off errors. However, Z_{bus}, which is required for short-circuit analysis, can be obtained directly by the method of *building algorithm* without the need for matrix inversion. This technique is discussed in Chapter 9.

Based on (6.3) and (6.4), the bus admittance matrix for the network in Figure 6.2 obtained by inspection is

$$Y_{bus} = \begin{bmatrix} -j8.50 & j2.50 & j5.00 & 0 \\ j2.50 & -j8.75 & j5.00 & 0 \\ j5.00 & j5.00 & -j22.50 & j12.50 \\ 0 & 0 & j12.50 & -j12.50 \end{bmatrix}.$$

A function called **Y = ybus(zdata)** is written for the formation of the bus admittance matrix. **zdata** is the line data input and contains four columns. The first two columns are the line bus numbers and the remaining columns contain the line resistance and reactance in per unit. The function returns the bus admittance matrix. The algorithm for the bus admittance program is very simple and basic to power system programming. Therefore, it is presented here for the reader to study and understand the method of solution. In the program, the line impedances are first converted to admittances. Y is then initialized to zero. In the first loop, the line data is searched, and the off-diagonal elements are entered. Finally, in a nested loop, line data is searched to find the elements connected to a bus, and the diagonal elements are thus formed.

The following is a program for building the bus admittance matrix:

```
function[Y] = ybus(zdata)
nl=zdata(:,1); nr=zdata(:,2); R=zdata(:,3); X=zdata(:,4);
nbr=length(zdata(:,1)); nbus = max(max(nl), max(nr));
Z = R + j*X;                                %branch impedance
```

```
y= ones(nbr,1)./Z;                          %branch admittance
Y = zeros(nbus,nbus);                      % initialize Y to zero
for k = 1:nbr;    % formation of the off diagonal elements
    if nl(k) > 0 & nr(k) > 0
    Y(nl(k),nr(k)) = Y(nl(k),nr(k)) - y(k);
    Y(nr(k),nl(k)) = Y(nl(k),nr(k));
    end
end
for n = 1:nbus        % formation of the diagonal elements
    for k = 1:nbr
    if nl(k) == n | nr(k) == n
    Y(n,n) = Y(n,n) + y(k);
    else, end
    end
end
```

Example 6.1

The emfs shown in Figure 6.1 are $E_1 = 1.1\angle 0°$ and $E_2 = 1.0\angle 0°$. Use the function **Y = ybus(zdata)** to obtain the bus admittance matrix. Find the bus impedance matrix by inversion, and solve for the bus voltages.

With source transformation, the equivalent current sources are

$$I_1 = \frac{1.1}{j1.0} = -j1.1 \text{ pu}$$
$$I_2 = \frac{1.0}{j0.8} = -j1.25 \text{ pu}$$

The following commands

```
%     From  To   R    X
z = [ 0     1    0    1.0
      0     2    0    0.8
      1     2    0    0.4
      1     3    0    0.2
      2     3    0    0.2
      3     4    0    0.08];
Y = ybus(z)                              % bus admittance matrix
Ibus = [-j*1.1; -j*1.25; 0; 0];        % vector of bus currents
Zbus = inv(Y)                            % bus impedance matrix
Vbus = Zbus*Ibus
```

result in

```
 Y  =
      0 - 8.50i    0 + 2.50i    0 +  5.00i    0 +  0.00i
      0 + 2.50i    0 - 8.75i    0 +  5.00i    0 +  0.00i
      0 + 5.00i    0 + 5.00i    0 - 22.50i    0 + 12.50i
      0 + 0.00i    0 + 0.00i    0 + 12.50i    0 - 12.50i
Zbus =
      0 + 0.50i    0 + 0.40i    0 + 0.450i    0 + 0.450i
      0 + 0.40i    0 + 0.48i    0 + 0.440i    0 + 0.440i
      0 + 0.45i    0 + 0.44i    0 + 0.545i    0 + 0.545i
      0 + 0.45i    0 + 0.44i    0 + 0.545i    0 + 0.625i
Vbus =
      1.0500
      1.0400
      1.0450
      1.0450
```

The solution of equation $\mathbf{I}_{bus} = \mathbf{Y}_{bus}\mathbf{V}_{bus}$ by inversion is very inefficient. It is not necessary to obtain the inverse of $\mathbf{Y}_{bus}$. Instead, direct solution is obtained by optimally ordered triangular factorization. In *MATLAB*, the solution of linear simultaneous equations $AX = B$ is obtained by using the matrix division operator \ (i.e., $X = A \setminus B$), which is based on the triangular factorization and Gaussian elimination. This technique is superior in both execution time and numerical accuracy. It is two to three times as fast and produces residuals on the order of machine accuracy.

In Example 6.1, obtain the direct solution by replacing the statements `Zbus = inv(Y)` and `Vbus = Zbus*Ibus` with `Vbus = Y\ Ibus`.

6.3 SOLUTION OF NONLINEAR ALGEBRAIC EQUATIONS

The most common techniques used for the iterative solution of nonlinear algebraic equations are Gauss-Seidel, Newton-Raphson, and Quasi-Newton methods. The Gauss-Seidel and Newton-Raphson methods are discussed for one-dimensional equation, and are then extended to n-dimensional equations.

6.3.1 GAUSS-SEIDEL METHOD

The Gauss-Seidel method is also known as the method of successive displacements. To illustrate the technique, consider the solution of the nonlinear equation given by

$$f(x) = 0 \tag{6.6}$$

The above function is rearranged and written as

$$x = g(x) \tag{6.7}$$

If $x^{(k)}$ is an initial estimate of the variable x, the following iterative sequence is formed.

$$x^{(k+1)} = g(x^{(k)}) \tag{6.8}$$

A solution is obtained when the difference between the absolute value of the successive iteration is less than a specified accuracy, i.e.,

$$|x^{(k+1)} - x^{(k)}| \leq \epsilon \tag{6.9}$$

where ϵ is the desired accuracy.

Example 6.2

Use the Gauss-Seidel method to find a root of the following equation

$$f(x) = x^3 - 6x^2 + 9x - 4 = 0$$

Solving for x, the above expression is written as

$$\begin{aligned} x &= -\frac{1}{9}x^3 + \frac{6}{9}x^2 + \frac{4}{9} \\ &= g(x) \end{aligned}$$

The *MATLAB* **plot** command is used to plot $g(x)$ and x over a range of 0 to 4.5, as shown in Figure 6.3. The intersections of $g(x)$ and x results in the roots of $f(x)$. From Figure 6.3 two of the roots are found to be 1 and 4. Actually, there is a repeated root at $x = 1$. Apply the Gauss-Seidel algorithm, and use an initial estimate of

$$x^{(0)} = 2$$

From (6.8), the first iteration is

$$x^{(1)} = g(2) = -\frac{1}{9}(2)^3 + \frac{6}{9}(2)^2 + \frac{4}{9} = 2.2222$$

The second iteration is

$$x^{(2)} = g(2.2222) = -\frac{1}{9}(2.2222)^3 + \frac{6}{9}(2.2222)^2 + \frac{4}{9} = 2.5173$$

The subsequent iterations result in 2.8966, 3.3376, 3.7398, 3.9568, 3.9988 and 4.0000. The process is repeated until the change in variable is within the desired

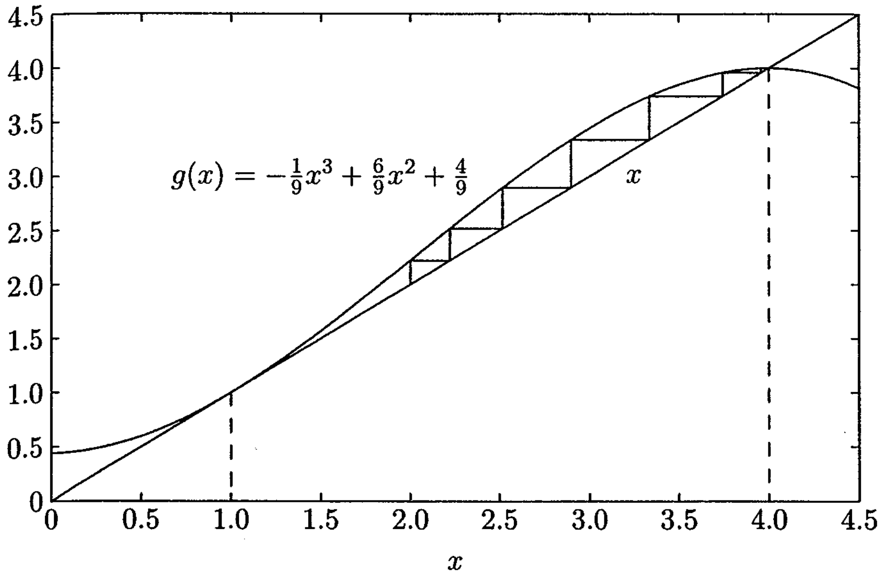

FIGURE 6.3
Graphical illustration of the Gauss-Seidel method.

accuracy. It can be seen that the Gauss-Seidel method needs many iterations to achieve the desired accuracy, and there is no guarantee for the convergence. In this example, since the initial estimate was within a "boxed in" region, the solution converged in a zigzag fashion to one of the roots. In fact, if the initial estimate was outside this region, say $x^{(0)} = 6$, the process would diverge. A test of convergence, especially for the n-dimensional case, is difficult, and no general methods are known.

The following commands show the procedure for the solution of the given equation starting with an initial estimate of $x^{(0)} = 2$.

```
dx=1;            % Change in variable is set to a high value
x=2;                                          % Initial estimate
iter = 0;                                    % Iteration counter
disp('Iter    g          dx          x')%Heading for results
while abs(dx) >= 0.001 & iter < 100 %Test for convergence
iter = iter + 1;                              % No. of iterations
g = -1/9*x^3+6/9*x^2+4/9 ;
dx = g-x;                                    % Change in variable
x = x + dx;                            % Successive approximation
fprintf('%g', iter), disp([g, dx, x])
end
```

The result is

Iter	g	dx	x
1	2.2222	0.2222	2.2222
2	2.5173	0.2951	2.5173
3	2.8966	0.3793	2.8966
4	3.3376	0.4410	3.3376
5	3.7398	0.4022	3.7398
6	3.9568	0.2170	3.9568
7	3.9988	0.0420	3.9988
8	4.0000	0.0012	4.0000
9	4.0000	0.0000	4.0000

In some cases, an acceleration factor can be used to improve the rate of convergence. If $\alpha > 1$ is the acceleration factor, the Gauss-Seidel algorithm becomes

$$x^{(k+1)} = x^{(k)} + \alpha[g(x^{(k)}) - x^{(k)}] \tag{6.10}$$

Example 6.3

Find a root of the equation in Example 6.2, using the Gauss-Seidel method with an acceleration factor of $\alpha = 1.25$:

Starting with an initial estimate of $x^{(0)} = 2$ and using (6.10), the first iteration is

$$\begin{aligned} g(2) &= -\frac{1}{9}(2)^3 + \frac{6}{9}(2)^2 + \frac{4}{9} = 2.2222 \\ x^{(1)} &= 2 + 1.25[2.2222 - 2] = 2.2778 \end{aligned}$$

The second iteration is

$$\begin{aligned} g(2.2778) &= -\frac{1}{9}(2.2778)^3 + \frac{6}{9}(2.2778)^2 + \frac{4}{9} = 2.5902 \\ x^{(2)} &= 2.2778 + 1.25[2.5902 - 2.2778] = 2.6683 \end{aligned}$$

The subsequent iterations result in 3.0801, 3.1831, 3.7238, 4.0084, 3.9978 and 4.0005. The effect of acceleration is shown graphically in Figure 6.4. Care must be taken not to use a very large acceleration factor since the larger step size may result in an overshoot. This can cause an increase in the number of iterations or even result in divergence. In the *MATLAB* command of Example 6.2, replace the command before the end statement by $x = x + 1.25 * dx$ to reflect the effect of the acceleration factor and run the program.

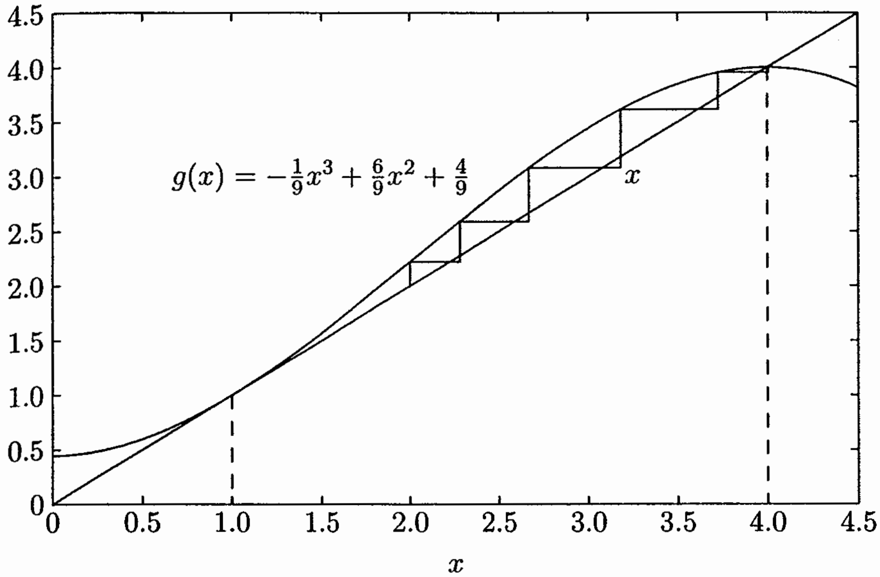

FIGURE 6.4
Graphical illustration of the Gauss-Seidel method using acceleration factor.

We now consider the system of n equations in n variables

$$\begin{aligned} f_1(x_1, x_2, \cdots, x_n) &= c_1 \\ f_2(x_1, x_2, \cdots, x_n) &= c_2 \\ &\cdots\cdots \\ f_n(x_1, x_2, \cdots, x_n) &= c_n \end{aligned} \tag{6.11}$$

Solving for one variable from each equation, the above functions are rearranged and written as

$$\begin{aligned} x_1 &= c_1 + g_1(x_1, x_2, \cdots, x_n) \\ x_2 &= c_2 + g_2(x_1, x_2, \cdots, x_n) \\ &\cdots\cdots \\ x_n &= c_n + g_n(x_1, x_2, \cdots, x_n) \end{aligned} \tag{6.12}$$

The iteration procedure is initiated by assuming an approximate solution for each of the independent variables $(x_1^{(0)}, x_2^{(0)} \cdots, x_n^{(0)})$. Equation (6.12) results in a new approximate solution $(x_1^{(1)}, x_2^{(1)} \cdots, x_n^{(1)})$. In the Gauss-Seidel method, the updated values of the variables calculated in the preceding equations are immediately used in the solution of the subsequent equations. At the end of each iteration, the calculated values of all variables are tested against the previous values. If all changes

in the variables are within the specified accuracy, a solution has converged, otherwise another iteration must be performed. The rate of convergence can often be increased by using a suitable acceleration factor α, and the iterative sequence becomes

$$x_i^{(k+1)} = x_i^{(k)} + \alpha(x_{i\,cal}^{(k+1)} - x_i^{(k)}) \tag{6.13}$$

6.3.2 NEWTON-RAPHSON METHOD

The most widely used method for solving simultaneous nonlinear algebraic equations is the Newton-Raphson method. Newton's method is a successive approximation procedure based on an initial estimate of the unknown and the use of Taylor's series expansion. Consider the solution of the one-dimensional equation given by

$$f(x) = c \tag{6.14}$$

If $x^{(0)}$ is an initial estimate of the solution, and $\Delta x^{(0)}$ is a small deviation from the correct solution, we must have

$$f(x^{(0)} + \Delta x^{(0)}) = c$$

Expanding the left-hand side of the above equation in Taylor's series about $x^{(0)}$ yields

$$f(x^{(0)}) + \left(\frac{df}{dx}\right)^{(0)} \Delta x^{(0)} + \frac{1}{2!}\left(\frac{d^2 f}{dx^2}\right)^{(0)} (\Delta x^{(0)})^2 + \cdots = c$$

Assuming the error $\Delta x^{(0)}$ is very small, the higher-order terms can be neglected, which results in

$$\Delta c^{(0)} \simeq \left(\frac{df}{dx}\right)^{(0)} \Delta x^{(0)}$$

where

$$\Delta c^{(0)} = c - f(x^{(0)})$$

Adding $\Delta x^{(0)}$ to the initial estimate will result in the second approximation

$$x^{(1)} = x^{(0)} + \frac{\Delta c^{(0)}}{\left(\frac{df}{dx}\right)^{(0)}}$$

Successive use of this procedure yields the Newton-Raphson algorithm

$$\Delta c^{(k)} = c - f(x^{(k)}) \tag{6.15}$$

$$\Delta x^{(k)} = \frac{\Delta c^{(k)}}{\left(\frac{df}{dx}\right)^{(k)}} \tag{6.16}$$

$$x^{(k+1)} = x^{(k)} + \Delta x^{(k)} \tag{6.17}$$

(6.16) can be rearranged as

$$\Delta c^{(k)} = j^{(k)} \Delta x^{(k)} \tag{6.18}$$

where

$$j^{(k)} = \left(\frac{df}{dx}\right)^{(k)}$$

The relation in (6.18) demonstrates that the nonlinear equation $f(x) - c = 0$ is approximated by the tangent line on the curve at $x^{(k)}$. Therefore, a linear equation is obtained in terms of the small changes in the variable. The intersection of the tangent line with the x-axis results in $x^{(k+1)}$. This idea is demonstrated graphically in Example 6.4.

Example 6.4

Use the Newton-Raphson method to find a root of the equation given in Example 6.2. Assume an initial estimate of $x^{(0)} = 6$.

The *MATLAB* **plot** command is used to plot $f(x) = x^3 - 6x^2 + 9x - 4$ over a range of 0 to 6 as shown in Figure 6.5. The intersections of $f(x)$ with the x-axis results in the roots of $f(x)$. From Figure 6.5, two of the roots are found to be 1 and 4. Actually, there is a repeated root at $x = 1$.

Also, Figure 6.5 gives a graphical description of the Newton-Raphson method. Starting with an initial estimate of $x^{(0)} = 6$, we extrapolate along the tangent to its intersection with the x-axis and take that as the next approximation. This is continued until successive x-values are sufficiently close.

The analytical solution given by the Newton-Raphson algorithm is

$$\frac{df(x)}{dx} = 3x^2 - 12x + 9$$

$$\Delta c^{(0)} = c - f(x^{(0)}) = 0 - [(6)^3 - 6(6)^2 + 9(6) - 4] = -50$$

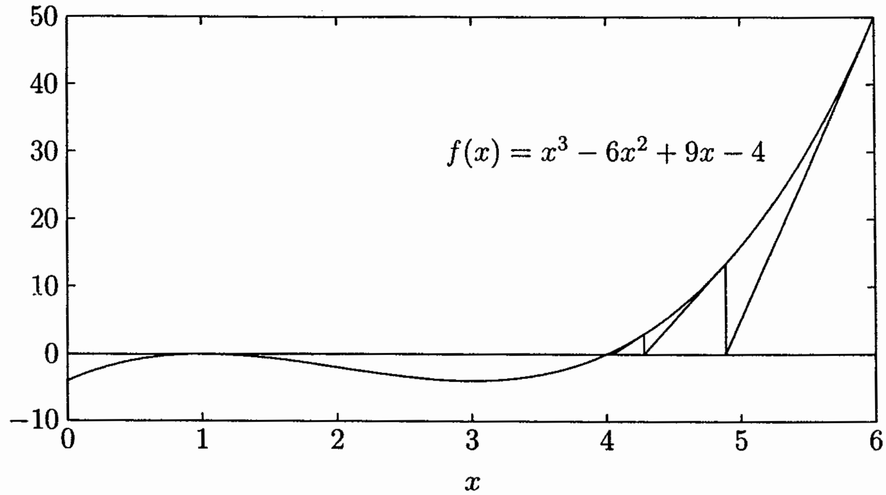

FIGURE 6.5
Graphical illustration of the Newton-Raphson algorithm.

$$\left(\frac{df}{dx}\right)^{(0)} = 3(6)^2 - 12(6) + 9 = 45$$

$$\Delta x^{(0)} = \frac{\Delta c^{(0)}}{\left(\frac{df}{dx}\right)^{(0)}} = \frac{-50}{45} = -1.1111$$

Therefore, the result at the end of the first iteration is

$$x^{(1)} = x^{(0)} + \Delta x^{(0)} = 6 - 1.1111 = 4.8889$$

The subsequent iterations result in

$$x^{(2)} = x^{(1)} + \Delta x^{(1)} = 4.8889 - \frac{13.4431}{22.037} = 4.2789$$

$$x^{(3)} = x^{(2)} + \Delta x^{(2)} = 4.2789 - \frac{2.9981}{12.5797} = 4.0405$$

$$x^{(4)} = x^{(3)} + \Delta x^{(3)} = 4.0405 - \frac{0.3748}{9.4914} = 4.0011$$

$$x^{(5)} = x^{(4)} + \Delta x^{(4)} = 4.0011 - \frac{0.0095}{9.0126} = 4.0000$$

We see that Newton's method converges considerably more rapidly than the Gauss-Seidel method. The method may converge to a root different from the expected one or diverge if the starting value is not close enough to the root.

The following commands show the procedure for the solution of the given equation by the Newton-Raphson method.

```
dx=1;          % Change in variable is set to a high value
x=input('Enter initial estimate -> '); % Initial estimate
iter = 0;                               % Iteration counter
disp('iter    Dc          J          dx           x') % Heading
while abs(dx) >= 0.001 & iter < 100  Test for convergence
iter = iter + 1;                        % No. of iterations
Dc = 0 - (x^3 - 6*x^2 + 9*x - 4);              % Residual
J = 3*x^2 - 12*x + 9;                        % Derivative
dx= Dc/J;                               %Change in variable
x=x + dx;                             % Successive solution
fprintf('%g', iter), disp([Dc, J, dx, x])
end
```

The result is

```
Enter the initial estimate -> 6
iter   Dc          J          dx          x
 1  -50.0000   45.0000   -1.1111     4.8889
 2  -13.4431   22.0370   -0.6100     4.2789
 3   -2.9981   12.5797   -0.2383     4.0405
 4   -0.3748    9.4914   -0.0395     4.0011
 5   -0.0095    9.0126   -0.0011     4.0000
 6   -0.0000    9.0000   -0.0000     4.0000
```

Now consider the n-dimensional equations given by (6.11). Expanding the left-hand side of the equations (6.11) in the Taylor's series about the initial estimates and neglecting all higher order terms, leads to the expression

$$
\begin{aligned}
(f_1)^{(0)} + \left(\frac{\partial f_1}{\partial x_1}\right)^{(0)} \Delta x_1^{(0)} + \left(\frac{\partial f_1}{\partial x_2}\right)^{(0)} \Delta x_2^{(0)} + \cdots + \left(\frac{\partial f_1}{\partial x_n}\right)^{(0)} \Delta x_n^{(0)} &= c_1 \\
(f_2)^{(0)} + \left(\frac{\partial f_2}{\partial x_1}\right)^{(0)} \Delta x_1^{(0)} + \left(\frac{\partial f_2}{\partial x_2}\right)^{(0)} \Delta x_2^{(0)} + \cdots + \left(\frac{\partial f_2}{\partial x_n}\right)^{(0)} \Delta x_n^{(0)} &= c_2 \\
&\vdots \\
(f_n)^{(0)} + \left(\frac{\partial f_n}{\partial x_1}\right)^{(0)} \Delta x_1^{(0)} + \left(\frac{\partial f_n}{\partial x_2}\right)^{(0)} \Delta x_2^{(0)} + \cdots + \left(\frac{\partial f_n}{\partial x_n}\right)^{(0)} \Delta x_n^{(0)} &= c_n
\end{aligned}
$$

or in matrix form

$$\begin{bmatrix} c_1 - (f_1)^{(0)} \\ c_2 - (f_2)^{(0)} \\ \vdots \\ c_n - (f_n)^{(0)} \end{bmatrix} = \begin{bmatrix} \left(\frac{\partial f_1}{\partial x_1}\right)^{(0)} & \left(\frac{\partial f_1}{\partial x_2}\right)^{(0)} & \cdots & \left(\frac{\partial f_1}{\partial x_n}\right)^{(0)} \\ \left(\frac{\partial f_2}{\partial x_1}\right)^{(0)} & \left(\frac{\partial f_2}{\partial x_2}\right)^{(0)} & \cdots & \left(\frac{\partial f_2}{\partial x_n}\right)^{(0)} \\ \vdots & \vdots & \ddots & \cdots \\ \left(\frac{\partial f_n}{\partial x_1}\right)^{(0)} & \left(\frac{\partial f_n}{\partial x_2}\right)^{(0)} & \cdots & \left(\frac{\partial f_n}{\partial x_n}\right)^{(0)} \end{bmatrix} \begin{bmatrix} \Delta x_1^{(0)} \\ \Delta x_2^{(0)} \\ \vdots \\ \Delta x_n^{(0)} \end{bmatrix}$$

In short form, it can be written as

$$\Delta C^{(k)} = J^{(k)} \Delta X^{(k)}$$

or

$$\Delta X^{(k)} = [J^{(k)}]^{-1} \Delta C^{(k)} \tag{6.19}$$

and the Newton-Raphson algorithm for the n-dimensional case becomes

$$X^{(k+1)} = X^{(k)} + \Delta X^{(k)} \tag{6.20}$$

where

$$\Delta X^{(k)} = \begin{bmatrix} \Delta x_1^{(k)} \\ \Delta x_2^{(k)} \\ \vdots \\ \Delta x_n^{(k)} \end{bmatrix} \quad \text{and} \quad \Delta C^{(k)} = \begin{bmatrix} c_1 - (f_1)^{(k)} \\ c_2 - (f_2)^{(k)} \\ \vdots \\ c_n - (f_n)^{(k)} \end{bmatrix} \tag{6.21}$$

$$J^{(k)} = \begin{bmatrix} \left(\frac{\partial f_1}{\partial x_1}\right)^{(k)} & \left(\frac{\partial f_1}{\partial x_2}\right)^{(k)} & \cdots & \left(\frac{\partial f_1}{\partial x_n}\right)^{(k)} \\ \left(\frac{\partial f_2}{\partial x_1}\right)^{(k)} & \left(\frac{\partial f_2}{\partial x_2}\right)^{(k)} & \cdots & \left(\frac{\partial f_2}{\partial x_n}\right)^{(k)} \\ \vdots & \vdots & \ddots & \cdots \\ \left(\frac{\partial f_n}{\partial x_1}\right)^{(k)} & \left(\frac{\partial f_n}{\partial x_2}\right)^{(k)} & \cdots & \left(\frac{\partial f_n}{\partial x_n}\right)^{(k)} \end{bmatrix} \tag{6.22}$$

$J^{(k)}$ is called the *Jacobian matrix*. Elements of this matrix are the partial derivatives evaluated at $X^{(k)}$. It is assumed that $J^{(k)}$ has an inverse during each iteration. Newton's method, as applied to a set of nonlinear equations, reduces the problem to solving a set of linear equations in order to determine the values that improve the accuracy of the estimates.

The solution of (6.19) by inversion is very inefficient. It is not necessary to obtain the inverse of $J^{(k)}$. Instead, a direct solution is obtained by optimally ordered triangular factorization. In *MATLAB*, the solution of linear simultaneous equations $\Delta C = J\Delta X$ is obtained by using the matrix division operator \ (i.e., $\Delta X = J \setminus \Delta C$) which is based on the triangular factorization and Gaussian elimination.

Example 6.5

Use the Newton-Raphson method to find the intersections of the curves

$$\begin{aligned} x_1^2 + x_2^2 &= 4 \\ e^{x_1} + x_2 &= 1 \end{aligned}$$

Graphically, the solution to this system is represented by the intersections of the circle $x_1^2 + x_2^2 = 4$ with the curve $e^{x_1} + x_2 = 1$. Figure 6.6 shows that these are near $(1, -1.7)$ and $(-1.8, 0.8)$.

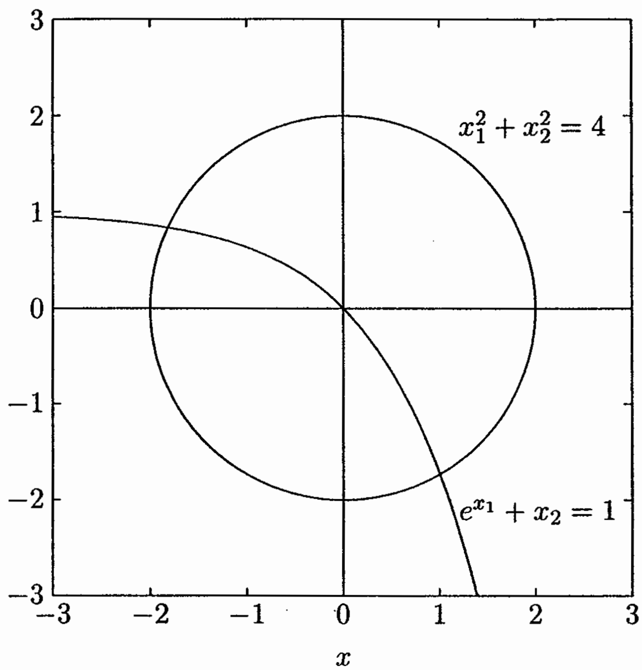

FIGURE 6.6
Graphs of Example 6.5.

Taking partial derivatives of the above functions results in the Jacobian matrix

$$J = \begin{bmatrix} 2x_1 & 2x_2 \\ e^{x_1} & 1 \end{bmatrix}$$

The Newton-Raphson algorithm for the above system is presented in the following statements.

```
iter = 0;                                   % Iteration counter
```

```
x=input('Enter initial estimates, col. vector[x1;x2]->');
Dx = [1; 1];  % Change in variable is set to a high value
C=[4; 1];
disp('Iter   DC       Jacobian matrix        Dx         x');
                                          % Heading for results
while max(abs(Dx)) >= 0.0001 & iter <10 %Convergence test
iter=iter+1;                               % Iteration counter
f = [x(1)^2+x(2)^2;  exp(x(1))+x(2)];               % Functions
DC = C - f;                                         % Residuals
J = [2*x(1)      2*x(2)                       % Jacobian matrix
     exp(x(1))        1];
Dx=J\DC;                                  % Change in variables
x=x+Dx;                                  % Successive solutions
fprintf('%g', iter), disp([DC, J, Dx, x])           % Results
end
```

When the program is run, the user is prompted to enter the initial estimate. Let us try an initial estimate given by [0.5; -1].

Enter Initial estimates, col. vector $[x_1;\ x_2] \rightarrow [0.5;\ -1]$

Iter	ΔC	Jacobian matrix		Δx	x
1	2.7500	1.0000	-2.0000	0.8034	1.3034
	0.3513	1.6487	1.0000	-0.9733	-1.9733
2	-1.5928	2.6068	-3.9466	-0.2561	1.0473
	-0.7085	3.6818	1.0000	0.2344	-1.7389
3	-0.1205	2.0946	-3.4778	-0.0422	1.0051
	-0.1111	2.8499	1.0000	0.0092	-1.7296
4	-0.0019	2.0102	-3.4593	-0.0009	1.0042
	-0.0025	2.7321	1.0000	0.0000	-1.7296
5	-0.0000	2.0083	-3.4593	-0.0000	1.0042
	-0.0000	2.7296	1.0000	-0.0000	-1.7296

After five iterations, the solution converges to $x_1 = 1.0042$ and $x_2 = -1.7296$ accurate to four decimal places. Starting with an initial value of [−0.5; 1], which is closer to the other intersection, results in $x_1 = -1.8163$ and $x_2 = 0.8374$.

Example 6.6

Starting with the initial values, $x_1 = 1$, $x_2 = 1$, and $x_3 = 1$, solve the following system of equations by the Newton-Raphson method.

$$\begin{aligned} x_1^2 - x_2^2 + x_3^2 &= 11 \\ x_1x_2 + x_2^2 - 3x_3 &= 3 \\ x_1 - x_1x_3 + x_2x_3 &= 6 \end{aligned}$$

Taking partial derivatives of the above functions results in the Jacobian matrix

$$J = \begin{bmatrix} 2x_1 & -2x_2 & 2x_3 \\ x_2 & x_1 + 2x_2 & -3 \\ 1 - x_3 & x_3 & -x_1 + x_2 \end{bmatrix}$$

The following statements solve the given system of equations by the Newton-Raphson algorithm

```
Dx=[10;10;10]; %Change in variable is set to a high value
x=[1; 1; 1];                                % Initial estimate
C=[11; 3; 6];
iter = 0;                                   % Iteration counter
while max(abs(Dx))>=.0001 & iter<10;%Test for convergence
iter = iter + 1                             % No. of iterations
F = [x(1)^2-x(2)^2+x(3)^2                       % Functions
     x(1)*x(2)+x(2)^2-3*x(3)
     x(1)-x(1)*x(3)+x(2)*x(3)];
DC =C - F                                       % Residuals
J = [2*x(1)  -2*x(2)      2*x(3)          % Jacobian matrix
     x(2)    x(1)+2*x(2)  -3
     1-x(3)  x(3)         -x(1)+x(2)]
Dx=J\DC                                     %Change in variable
x=x+Dx                                    % Successive solution
end
```

The program results for the first iteration are

```
DC =                 J =
     10                   2    -2     2
      4                   1     3    -3
      5                   0     1     0
Dx =                 x =
     4.750                5.750
     5.000                6.000
     5.250                6.250
```

After six iterations, the solution converges to $x_1 = 2.0000$, $x_2 = 3.0000$, and $x_3 = 4.0000$.

Newton's method has the advantage of converging quadratically when we are near a root. However, more functional evaluations are required during each iteration. A very important limitation is that it does not generally converge to a solution from an arbitrary starting point.

6.4 POWER FLOW SOLUTION

Power flow studies, commonly known as *load flow*, form an important part of power system analysis. They are necessary for planning, economic scheduling, and control of an existing system as well as planning its future expansion. The problem consists of determining the magnitudes and phase angle of voltages at each bus and active and reactive power flow in each line.

In solving a power flow problem, the system is assumed to be operating under balanced conditions and a single-phase model is used. Four quantities are associated with each bus. These are voltage magnitude $|V|$, phase angle δ, real power P, and reactive power Q. The system buses are generally classified into three types.

Slack bus One bus, known as *slack* or *swing bus*, is taken as reference where the magnitude and phase angle of the voltage are specified. This bus makes up the difference between the scheduled loads and generated power that are caused by the losses in the network.

Load buses At these buses the active and reactive powers are specified. The magnitude and the phase angle of the bus voltages are unknown. These buses are called P-Q buses.

Regulated buses These buses are the *generator buses*. They are also known as *voltage-controlled buses*. At these buses, the real power and voltage magnitude are specified. The phase angles of the voltages and the reactive power are to be determined. The limits on the value of the reactive power are also specified. These buses are called P-V buses.

6.4.1 POWER FLOW EQUATION

Consider a typical bus of a power system network as shown in Figure 6.7. Transmission lines are represented by their equivalent π models where impedances have been converted to per unit admittances on a common MVA base.

Application of KCL to this bus results in

$$\begin{aligned} I_i &= y_{i0}V_i + y_{i1}(V_i - V_1) + y_{i2}(V_i - V_2) + \cdots + y_{in}(V_i - V_n) \\ &= (y_{i0} + y_{i1} + y_{i2} + \cdots + y_{in})V_i - y_{i1}V_1 - y_{i2}V_2 - \cdots - y_{in}V_n \end{aligned} \tag{6.23}$$

or

$$I_i = V_i \sum_{j=0}^{n} y_{ij} - \sum_{j=1}^{n} y_{ij}V_j \qquad j \neq i \tag{6.24}$$

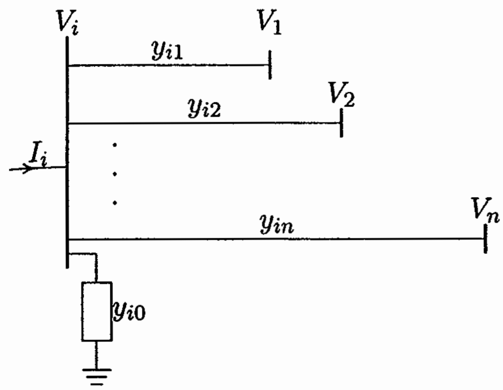

FIGURE 6.7
A typical bus of the power system.

The real and reactive power at bus i is

$$P_i + jQ_i = V_i I_i^* \tag{6.25}$$

or

$$I_i = \frac{P_i - jQ_i}{V_i^*} \tag{6.26}$$

Substituting for I_i in (6.24) yields

$$\frac{P_i - jQ_i}{V_i^*} = V_i \sum_{j=0}^{n} y_{ij} - \sum_{j=1}^{n} y_{ij} V_j \qquad j \neq i \tag{6.27}$$

From the above relation, the mathematical formulation of the power flow problem results in a system of algebraic nonlinear equations which must be solved by iterative techniques.

6.5 GAUSS-SEIDEL POWER FLOW SOLUTION

In the power flow study, it is necessary to solve the set of nonlinear equations represented by (6.27) for two unknown variables at each node. In the Gauss-Seidel method (6.27) is solved for V_i, and the iterative sequence becomes

$$V_i^{(k+1)} = \frac{\frac{P_i^{sch} - jQ_i^{sch}}{V_i^{*(k)}} + \sum y_{ij} V_j^{(k)}}{\sum y_{ij}} \qquad j \neq i \tag{6.28}$$

where y_{ij} shown in lowercase letters is the actual admittance in per unit. P_i^{sch} and Q_i^{sch} are the net real and reactive powers expressed in per unit. In writing the KCL, current entering bus i was assumed positive. Thus, for buses where real and reactive powers are injected into the bus, such as generator buses, P_i^{sch} and Q_i^{sch} have positive values. For load buses where real and reactive powers are flowing away from the bus, P_i^{sch} and Q_i^{sch} have negative values. If (6.27) is solved for P_i and Q_i, we have

$$P_i^{(k+1)} = \Re\{V_i^{*(k)}[V_i^{(k)}\sum_{j=0}^{n} y_{ij} - \sum_{j=1}^{n} y_{ij}V_j^{(k)}]\} \qquad j \neq i \tag{6.29}$$

$$Q_i^{(k+1)} = -\Im\{V_i^{*(k)}[V_i^{(k)}\sum_{j=0}^{n} y_{ij} - \sum_{j=1}^{n} y_{ij}V_j^{(k)}]\} \qquad j \neq i \tag{6.30}$$

The power flow equation is usually expressed in terms of the elements of the bus admittance matrix. Since the off-diagonal elements of the bus admittance matrix Y_{bus}, shown by uppercase letters, are $Y_{ij} = -y_{ij}$, and the diagonal elements are $Y_{ii} = \sum y_{ij}$, (6.28) becomes

$$V_i^{(k+1)} = \frac{\frac{P_i^{sch} - jQ_i^{sch}}{V_i^{*(k)}} - \sum_{j \neq i} Y_{ij}V_j^{(k)}}{Y_{ii}} \tag{6.31}$$

and

$$P_i^{(k+1)} = \Re\{V_i^{*(k)}[V_i^{(k)}Y_{ii} + \sum_{\substack{j=1 \\ j \neq i}}^{n} Y_{ij}V_j^{(k)}]\} \quad j \neq i \tag{6.32}$$

$$Q_i^{(k+1)} = -\Im\{V_i^{*(k)}[V_i^{(k)}Y_{ii} + \sum_{\substack{j=1 \\ j \neq i}}^{n} Y_{ij}V_j^{(k)}]\} \quad j \neq i \tag{6.33}$$

Y_{ii} includes the admittance to ground of line charging susceptance and any other fixed admittance to ground. In Section 6.7, a model is presented for transformers containing off-nominal ratio, which includes the effect of transformer tap setting.

Since both components of voltage are specified for the slack bus, there are $2(n-1)$ equations which must be solved by an iterative method. Under normal operating conditions, the voltage magnitude of buses are in the neighborhood of 1.0 per unit or close to the voltage magnitude of the slack bus. Voltage magnitude at load buses are somewhat lower than the slack bus value, depending on the reactive power demand, whereas the scheduled voltage at the generator buses are somewhat higher. Also, the phase angle of the load buses are below the reference angle in accordance to the real power demand, whereas the phase angle of the generator

buses may be above the reference value depending on the amount of real power flowing into the bus. Thus, for the Gauss-Seidel method, an initial voltage estimate of $1.0 + j0.0$ for unknown voltages is satisfactory, and the converged solution correlates with the actual operating states.

For P-Q buses, the real and reactive powers P_i^{sch} and Q_i^{sch} are known. Starting with an initial estimate, (6.31) is solved for the real and imaginary components of voltage. For the voltage-controlled buses (P-V buses) where P_i^{sch} and $|V_i|$ are specified, first (6.33) is solved for $Q_i^{(k+1)}$, and then is used in (6.31) to solve for $V_i^{(k+1)}$. However, since $|V_i|$ is specified, only the imaginary part of $V_i^{(k+1)}$ is retained, and its real part is selected in order to satisfy

$$(e_i^{(k+1)})^2 + (f_i^{(k+1)})^2 = |V_i|^2 \tag{6.34}$$

or

$$e_i^{(k+1)} = \sqrt{|V_i|^2 - (f_i^{(k+1)})^2} \tag{6.35}$$

where $e_i^{(k+1)}$ and $f_i^{(k+1)}$ are the real and imaginary components of the voltage $V_i^{(k+1)}$ in the iterative sequence.

The rate of convergence is increased by applying an acceleration factor to the approximate solution obtained from each iteration.

$$V_i^{(k+1)} = V_i^{(k)} + \alpha(V_{i\,cal}^{(k)} - V_i^{(k)}) \tag{6.36}$$

where α is the acceleration factor. Its value depends upon the system. The range of 1.3 to 1.7 is found to be satisfactory for typical systems.

The updated voltages immediately replace the previous values in the solution of the subsequent equations. The process is continued until changes in the real and imaginary components of bus voltages between successive iterations are within a specified accuracy, i.e.,

$$\begin{aligned} |e_i^{(k+1)} - e_i^{(k)}| &\leq \epsilon \\ |f_i^{(k+1)} - f_i^{(k)}| &\leq \epsilon \end{aligned} \tag{6.37}$$

For the power mismatch to be reasonably small and acceptable, a very tight tolerance must be specified on both components of the voltage. A voltage accuracy in the range of 0.00001 to 0.00005 pu is satisfactory. In practice, the method for determining the completion of a solution is based on an accuracy index set up on the power mismatch. The iteration continues until the magnitude of the largest element in the ΔP and ΔQ columns is less than the specified value. A typical power mismatch accuracy is 0.001 pu

Once a solution is converged, the net real and reactive powers at the slack bus are computed from (6.32) and (6.33).

6.6 LINE FLOWS AND LOSSES

After the iterative solution of bus voltages, the next step is the computation of line flows and line losses. Consider the line connecting the two buses i and j in Figure 6.8. The line current I_{ij}, measured at bus i and defined positive in the direction

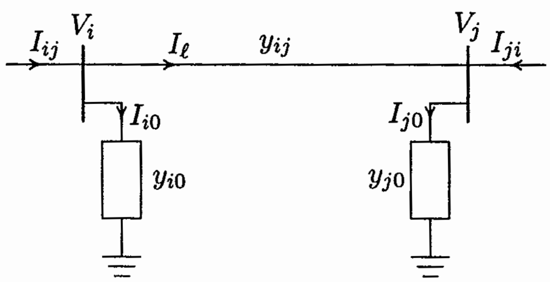

FIGURE 6.8
Transmission line model for calculating line flows.

$i \rightarrow j$ is given by

$$I_{ij} = I_\ell + I_{i0} = y_{ij}(V_i - V_j) + y_{i0}V_i \tag{6.38}$$

Similarly, the line current I_{ji} measured at bus j and defined positive in the direction $j \rightarrow i$ is given by

$$I_{ji} = -I_\ell + I_{j0} = y_{ij}(V_j - V_i) + y_{j0}V_j \tag{6.39}$$

The complex powers S_{ij} from bus i to j and S_{ji} from bus j to i are

$$S_{ij} = V_i I_{ij}^* \tag{6.40}$$

$$S_{ji} = V_j I_{ji}^* \tag{6.41}$$

The power loss in line $i - j$ is the algebraic sum of the power flows determined from (6.40) and (6.41), i.e.,

$$S_{L\,ij} = S_{ij} + S_{ji} \tag{6.42}$$

The power flow solution by the Gauss-Seidel method is demonstrated in the following two examples.

Example 6.7

Figure 6.9 shows the one-line diagram of a simple three-bus power system with generation at bus 1. The magnitude of voltage at bus 1 is adjusted to 1.05 per

unit. The scheduled loads at buses 2 and 3 are as marked on the diagram. Line impedances are marked in per unit on a 100-MVA base and the line charging susceptances are neglected.

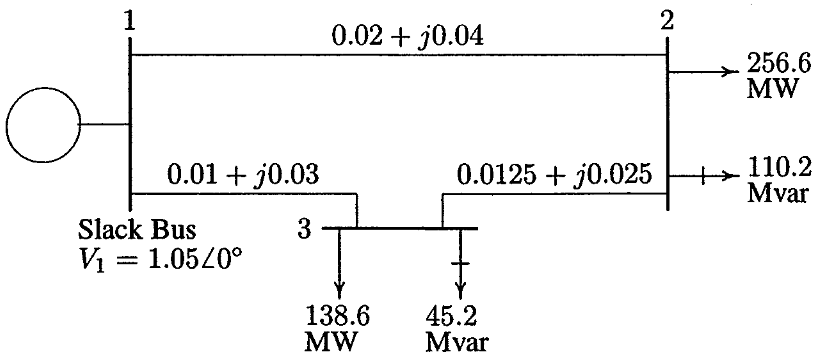

FIGURE 6.9
One-line diagram of Example 6.7 (impedances in pu on 100-MVA base).

(a) Using the Gauss-Seidel method, determine the phasor values of the voltage at the load buses 2 and 3 (P-Q buses) accurate to four decimal places.
(b) Find the slack bus real and reactive power.
(c) Determine the line flows and line losses. Construct a power flow diagram showing the direction of line flow.

(a) Line impedances are converted to admittances

$$y_{12} = \frac{1}{0.02 + j0.04} = 10 - j20$$

Similarly, $y_{13} = 10 - j30$ and $y_{23} = 16 - j32$. The admittances are marked on the network shown in Figure 6.10.
At the P-Q buses, the complex loads expressed in per units are

$$S_2^{sch} = -\frac{(256.6 + j110.2)}{100} = -2.566 - j1.102 \quad \text{pu}$$

$$S_3^{sch} = -\frac{(138.6 + j45.2)}{100} = -1.386 - j0.452 \quad \text{pu}$$

Since the actual admittances are readily available in Figure 6.10, for hand calculation, we use (6.28). Bus 1 is taken as reference bus (slack bus). Starting from an initial estimate of $V_2^{(0)} = 1.0 + j0.0$ and $V_3^{(0)} = 1.0 + j0.0$, V_2 and V_3 are computed from (6.28) as follows

$$V_2^{(1)} = \frac{\frac{P_2^{sch} - jQ_2^{sch}}{V_2^{*(0)}} + y_{12}V_1 + y_{23}V_3^{(0)}}{y_{12} + y_{23}}$$

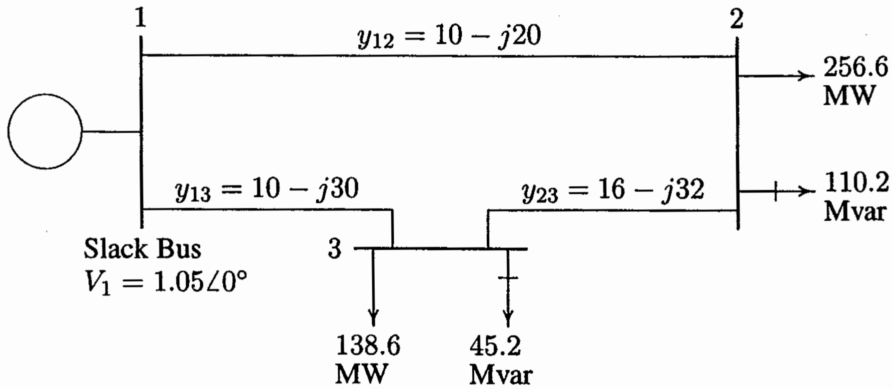

FIGURE 6.10
One-line diagram of Example 6.7 (admittances in pu on 100-MVA base).

$$= \frac{\frac{-2.566+j1.102}{1.0-j0} + (10-j20)(1.05+j0) + (16-j32)(1.0+j0)}{(26-j52)}$$
$$= 0.9825 - j0.0310$$

and

$$V_3^{(1)} = \frac{\frac{P_3^{sch}-jQ_3^{sch}}{V_3^{*(0)}} + y_{13}V_1 + y_{23}V_2^{(1)}}{y_{13}+y_{23}}$$
$$= \frac{\frac{-1.386+j0.452}{1-j0} + (10-j30)(1.05+j0) + (16-j32)(0.9825-j0.0310)}{(26-j62)}$$
$$= 1.0011 - j0.0353$$

For the second iteration we have

$$V_2^{(2)} = \frac{\frac{-2.566+j1.102}{0.9825+j0.0310} + (10-j20)(1.05+j0) + (16-j32)(1.0011-j0.0353)}{(26-j52)}$$
$$= 0.9816 - j0.0520$$

and

$$V_3^{(2)} = \frac{\frac{-1.386+j0.452}{1.0011+j0.0353} + (10-j30)(1.05+j0) + (16-j32)(0.9816-j0.052)}{(26-j62)}$$
$$= 1.0008 - j0.0459$$

The process is continued and a solution is converged with an accuracy of 5×10^{-5} per unit in seven iterations as given below.

$$V_2^{(3)} = 0.9808 - j0.0578 \qquad V_3^{(3)} = 1.0004 - j0.0488$$

$$V_2^{(4)} = 0.9803 - j0.0594 \qquad V_3^{(4)} = 1.0002 - j0.0497$$
$$V_2^{(5)} = 0.9801 - j0.0598 \qquad V_3^{(5)} = 1.0001 - j0.0499$$
$$V_2^{(6)} = 0.9801 - j0.0599 \qquad V_3^{(6)} = 1.0000 - j0.0500$$
$$V_2^{(7)} = 0.9800 - j0.0600 \qquad V_3^{(7)} = 1.0000 - j0.0500$$

The final solution is

$$V_2 = 0.9800 - j0.0600 = 0.98183\angle{-3.5035^\circ} \quad \text{pu}$$
$$V_3 = 1.0000 - j0.0500 = 1.00125\angle{-2.8624^\circ} \quad \text{pu}$$

(b) With the knowledge of all bus voltages, the slack bus power is obtained from (6.27)

$$\begin{aligned}
P_1 - jQ_1 &= V_1^*[V_1(y_{12} + y_{13}) - (y_{12}V_2 + y_{13}V_3)] \\
&= 1.05[1.05(20 - j50) - (10 - j20)(0.98 - j.06) - \\
&\quad (10 - j30)(1.0 - j0.05)] \\
&= 4.095 - j1.890
\end{aligned}$$

or the slack bus real and reactive powers are $P_1 = 4.095$ pu $= 409.5$ MW and $Q_1 = 1.890$ pu $= 189$ Mvar.

(c) To find the line flows, first the line currents are computed. With line charging capacitors neglected, the line currents are

$$I_{12} = y_{12}(V_1 - V_2) = (10 - j20)[(1.05 + j0) - (0.98 - j0.06)] = 1.9 - j0.8$$
$$I_{21} = -I_{12} = -1.9 + j0.8$$
$$I_{13} = y_{13}(V_1 - V_3) = (10 - j30)[(1.05 + j0) - (1.0 - j0.05)] = 2.0 - j1.0$$
$$I_{31} = -I_{13} = -2.0 + j1.0$$
$$I_{23} = y_{23}(V_2 - V_3) = (16 - j32)[(0.98 - j0.06) - (1 - j0.05)] = -.64 + j.48$$
$$I_{32} = -I_{23} = 0.64 - j0.48$$

The line flows are

$$\begin{aligned}
S_{12} = V_1 I_{12}^* &= (1.05 + j0.0)(1.9 + j0.8) = 1.995 + j0.84 \text{ pu} \\
&= 199.5 \text{ MW} + j84.0 \text{ Mvar} \\
S_{21} = V_2 I_{21}^* &= (0.98 - j0.06)(-1.9 - j0.8) = -1.91 - j0.67 \text{ pu} \\
&= -191.0 \text{ MW} - j67.0 \text{ Mvar} \\
S_{13} = V_1 I_{13}^* &= (1.05 + j0.0)(2.0 + j1.0) = 2.1 + j1.05 \text{ pu} \\
&= 210.0 \text{ MW} + j105.0 \text{ Mvar}
\end{aligned}$$

$$S_{31} = V_3 I_{31}^* = (1.0 - j0.05)(-2.0 - j1.0) = -2.05 - j0.90 \text{ pu}$$
$$= -205.0 \text{ MW} - j90.0 \text{ Mvar}$$
$$S_{23} = V_2 I_{23}^* = (0.98 - j0.06)(-0.656 + j0.48) = -0.656 - j0.432 \text{ pu}$$
$$= -65.6 \text{ MW} - j43.2 \text{ Mvar}$$
$$S_{32} = V_3 I_{32}^* = (1.0 - j0.05)(0.64 + j0.48) = 0.664 + j0.448 \text{ pu}$$
$$= 66.4 \text{ MW} + j44.8 \text{ Mvar}$$

and the line losses are

$$S_{L\,12} = S_{12} + S_{21} = 8.5 \text{ MW} + j17.0 \text{ Mvar}$$
$$S_{L\,13} = S_{13} + S_{31} = 5.0 \text{ MW} + j15.0 \text{ Mvar}$$
$$S_{L\,23} = S_{23} + S_{32} = 0.8 \text{ MW} + j1.60 \text{ Mvar}$$

The power flow diagram is shown in Figure 6.11, where real power direction is indicated by $\rightarrow$ and the reactive power direction is indicated by $\mapsto$. The values within parentheses are the real and reactive losses in the line.

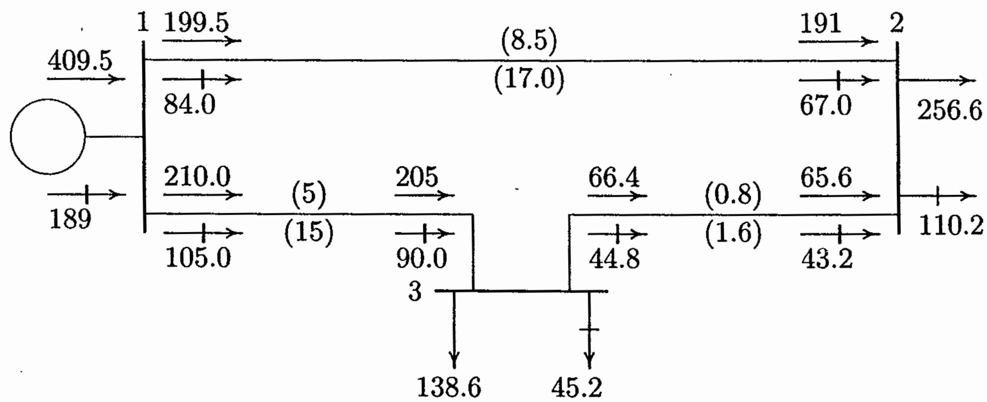

FIGURE 6.11
Power flow diagram of Example 6.7 (powers in MW and Mvar).

Example 6.8

Figure 6.12 shows the one-line diagram of a simple three-bus power system with generators at buses 1 and 3. The magnitude of voltage at bus 1 is adjusted to 1.05 pu. Voltage magnitude at bus 3 is fixed at 1.04 pu with a real power generation of 200 MW. A load consisting of 400 MW and 250 Mvar is taken from bus 2. Line impedances are marked in per unit on a 100 MVA base, and the line charging susceptances are neglected. Obtain the power flow solution by the Gauss-Seidel method including line flows and line losses.

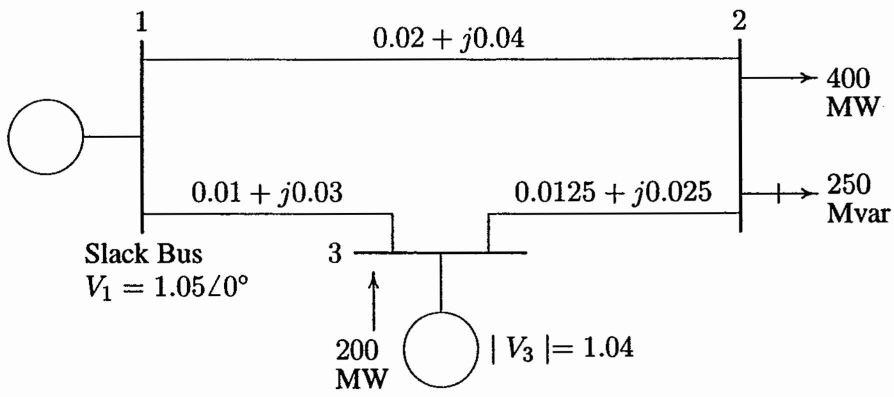

FIGURE 6.12
One-line diagram of Example 6.8 (impedances in pu on 100-MVA base).

Line impedances converted to admittances are $y_{12} = 10-j20$, $y_{13} = 10-j30$ and $y_{23} = 16 - j32$. The load and generation expressed in per units are

$$S_2^{sch} = -\frac{(400 + j250)}{100} = -4.0 - j2.5 \quad \text{pu}$$
$$P_3^{sch} = \frac{200}{100} = 2.0 \quad \text{pu}$$

Bus 1 is taken as the reference bus (slack bus). Starting from an initial estimate of $V_2^{(0)} = 1.0 + j0.0$ and $V_3^{(0)} = 1.04 + j0.0$, V_2 and V_3 are computed from (6.28).

$$\begin{aligned} V_2^{(1)} &= \frac{\frac{P_2^{sch} - jQ_2^{sch}}{V_2^{*(0)}} + y_{12}V_1 + y_{23}V_3^{(0)}}{y_{12} + y_{23}} \\ &= \frac{\frac{-4.0+j2.5}{1.0-j0} + (10 - j20)(1.05 + j0) + (16 - j32)(1.04 + j0)}{(26 - j52)} \\ &= 0.97462 - j0.042307 \end{aligned}$$

Bus 3 is a regulated bus where voltage magnitude and real power are specified. For the voltage-controlled bus, first the reactive power is computed from (6.30)

$$\begin{aligned} Q_3^{(1)} &= -\Im\{V_3^{*(0)}[V_3^{(0)}(y_{13} + y_{23}) - y_{13}V_1 - y_{23}V_2^{(1)}]\} \\ &= -\Im\{(1.04 - j0)[(1.04 + j0)(26 - j62) - (10 - j30)(1.05 + j0) - \\ &\quad (16 - j32)(0.97462 - j0.042307)]\} \\ &= 1.16 \end{aligned}$$

The value of $Q_3^{(1)}$ is used as Q_3^{sch} for the computation of voltage at bus 3. The complex voltage at bus 3, denoted by $V_{c3}^{(1)}$, is calculated

$$
\begin{aligned}
V_{c3}^{(1)} &= \frac{\frac{P_3^{sch} - jQ_3^{sch}}{V_3^{*(0)}} + y_{13}V_1 + y_{23}V_2^{(1)}}{y_{13} + y_{23}} \\
&= \frac{\frac{2.0 - j1.16}{1.04 - j0} + (10 - j30)(1.05 + j0) + (16 - j32)(0.97462 - j0.042307)}{(26 - j62)} \\
&= 1.03783 - j0.005170
\end{aligned}
$$

Since $|V_3|$ is held constant at 1.04 pu, only the imaginary part of $V_{c3}^{(1)}$ is retained, i.e, $f_3^{(1)} = -0.005170$, and its real part is obtained from

$$
e_3^{(1)} = \sqrt{(1.04)^2 - (0.005170)^2} = 1.039987
$$

Thus

$$
V_3^{(1)} = 1.039987 - j0.005170
$$

For the second iteration, we have

$$
\begin{aligned}
V_2^{(2)} &= \frac{\frac{P_2^{sch} - jQ_2^{sch}}{V_2^{*(1)}} + y_{12}V_1 + y_{23}V_3^{(1)}}{y_{12} + y_{23}} \\
&= \frac{\frac{-4.0 + j2.5}{.97462 + j.042307} + (10 - j20)(1.05) + (16 - j32)(1.039987 + j0.005170)}{(26 - j52)} \\
&= 0.971057 - j0.043432
\end{aligned}
$$

$$
\begin{aligned}
Q_3^{(2)} &= -\Im\{V_3^{*(1)}[V_3^{(1)}(y_{13} + y_{23}) - y_{13}V_1 - y_{23}V_2^{(2)}]\} \\
&= -\Im\{(1.039987 + j0.005170)[(1.039987 - j0.005170)(26 - j62) - \\
&\quad (10 - j30)(1.05 + j0) - (16 - j32)(0.971057 - j0.043432)]\} \\
&= 1.38796
\end{aligned}
$$

$$
\begin{aligned}
V_{c3}^{(2)} &= \frac{\frac{P_3^{sch} - jQ_3^{sch}}{V_3^{*(1)}} + y_{13}V_1 + y_{23}V_2^{(2)}}{y_{13} + y_{23}} \\
&= \frac{\frac{2.0 - j1.38796}{1.039987 + j0.00517} + (10 - j30)(1.05) + (16 - j32)(.971057 - j.043432)}{(26 - j62)} \\
&= 1.03908 - j0.00730
\end{aligned}
$$

Since $|V_3|$ is held constant at 1.04 pu, only the imaginary part of $V_{c3}^{(2)}$ is retained, i.e, $f_3^{(2)} = -0.00730$, and its real part is obtained from

$$e_3^{(2)} = \sqrt{(1.04)^2 - (0.00730)^2} = 1.039974$$

or

$$V_3^{(2)} = 1.039974 - j0.00730$$

The process is continued and a solution is converged with an accuracy of 5×10^{-5} pu in seven iterations as given below.

$$\begin{array}{lll} V_2^{(3)} = 0.97073 - j0.04479 & Q_3^{(3)} = 1.42904 & V_3^{(3)} = 1.03996 - j0.00833 \\ V_2^{(4)} = 0.97065 - j0.04533 & Q_3^{(4)} = 1.44833 & V_3^{(4)} = 1.03996 - j0.00873 \\ V_2^{(5)} = 0.97062 - j0.04555 & Q_3^{(5)} = 1.45621 & V_3^{(5)} = 1.03996 - j0.00893 \\ V_2^{(6)} = 0.97061 - j0.04565 & Q_3^{(6)} = 1.45947 & V_3^{(6)} = 1.03996 - j0.00900 \\ V_2^{(7)} = 0.97061 - j0.04569 & Q_3^{(7)} = 1.46082 & V_3^{(7)} = 1.03996 - j0.00903 \end{array}$$

The final solution is

$$V_2 = 0.97168\angle{-2.6948^\circ} \quad \text{pu}$$

$$\begin{aligned} S_3 &= 2.0 + j1.4617 \quad \text{pu} \\ V_3 &= 1.04\angle{-.498^\circ} \quad \text{pu} \\ S_1 &= 2.1842 + j1.4085 \quad \text{pu} \end{aligned}$$

Line flows and line losses are computed as in Example 6.7, and the results expressed in MW and Mvar are

$$\begin{array}{lll} S_{12} = 179.36 + j118.734 & S_{21} = -170.97 - j101.947 & S_{L12} = 8.39 + j16.79 \\ S_{13} = 39.06 + j22.118 & S_{31} = -38.88 - j\,21.569 & S_{L13} = 0.18 + j0.548 \\ S_{23} = -229.03 - j148.05 & S_{32} = 238.88 + j167.746 & S_{L23} = 9.85 + j19.69 \end{array}$$

The power flow diagram is shown in Figure 6.13, where real power direction is indicated by $\rightarrow$ and the reactive power direction is indicated by $\mapsto$. The values within parentheses are the real and reactive losses in the line.

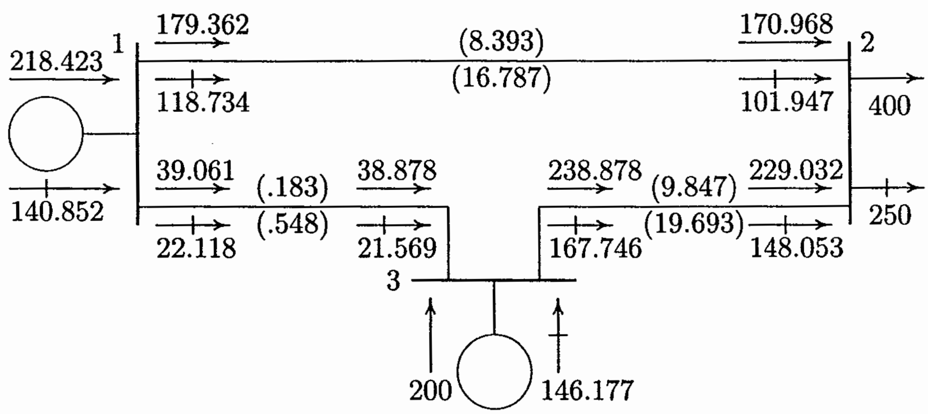

FIGURE 6.13
Power flow diagram of Example 6.8 (powers in MW and Mvar).

6.7 TAP CHANGING TRANSFORMERS

In Section 2.6 it was shown that the flow of real power along a transmission line is determined by the angle difference of the terminal voltages, and the flow of reactive power is determined mainly by the magnitude difference of terminal voltages. Real and reactive powers can be controlled by use of tap changing transformers and regulating transformers.

In a tap changing transformer, when the ratio is at the nominal value, the transformer is represented by a series admittance y_t in per unit. With off-nominal ratio, the per unit admittance is different from both sides of the transformer, and the admittance must be modified to include the effect of the off-nominal ratio. Consider a transformer with admittance y_t in series with an ideal transformer representing the off-nominal tap ratio 1:a as shown in Figure 6.14. y_t is the admittance in per unit based on the nominal turn ratio and a is the per unit off-nominal tap position allowing for small adjustment in voltage of usually ± 10 percent. In the case of phase shifting transformers, a is a complex number. Consider a fictitious bus x between the turn ratio and admittance of the transformer. Since the complex power on either side of the ideal transformer is the same, it follows that if the voltage goes through a positive phase angle shift, the current will go through a negative phase angle shift. Thus, for the assumed direction of currents, we have

$$V_x = \frac{1}{a}V_j \tag{6.43}$$

$$I_i = -a^* I_j \tag{6.44}$$

The current I_i is given by

$$I_i = y_t(V_i - V_x)$$

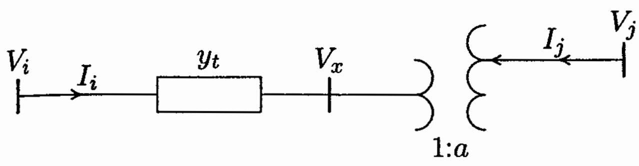

FIGURE 6.14
Transformer with tap setting ratio a:1

Substituting for V_x, we have

$$I_i = y_t V_i - \frac{y_t}{a} V_j \tag{6.45}$$

Also, from (6.44) we have

$$I_j = -\frac{1}{a^*} I_i$$

substituting for I_i from (6.45) we have

$$I_j = -\frac{y_t}{a^*} V_i + \frac{y_t}{|a|^2} V_j \tag{6.46}$$

writing (6.45) and (6.46) in matrix form results in

$$\begin{bmatrix} I_i \\ I_j \end{bmatrix} = \begin{bmatrix} y_t & -\frac{y_t}{a} \\ -\frac{y_t}{a^*} & \frac{y_t}{|a|^2} \end{bmatrix} \begin{bmatrix} V_i \\ V_j \end{bmatrix} \tag{6.47}$$

For the case when a is real, the π model shown in Figure 6.15 represents the admittance matrix in (6.47). In the π model, the left side corresponds to the non-tap side and the right side corresponds to the tap side of the transformer.

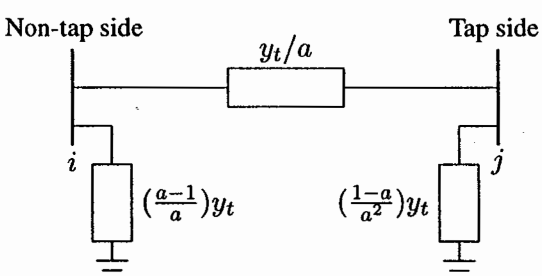

FIGURE 6.15
Equivalent circuit for a tap changing transformer.

6.8 POWER FLOW PROGRAMS

Several computer programs have been developed for the power flow solution of practical systems. Each method of solution consists of four programs. The program for the Gauss-Seidel method is **lfgauss**, which is preceded by **lfybus**, and is followed by **busout** and **lineflow**. Programs **lfybus**, **busout**, and **lineflow** are designed to be used with two more power flow programs. These are **lfnewton** for the Newton-Raphson method and **decouple** for the fast decoupled method. The following is a brief description of the programs used in the Gauss-Seidel method.

lfybus This program requires the line and transformer parameters and transformer tap settings specified in the input file named **linedata**. It converts impedances to admittances and obtains the bus admittance matrix. The program is designed to handle parallel lines.

lfgauss This program obtains the power flow solution by the Gauss-Seidel method and requires the files named **busdata** and **linedata**. It is designed for the direct use of load and generation in MW and Mvar, bus voltages in per unit, and angle in degrees. Loads and generation are converted to per unit quantities on the base MVA selected. A provision is made to maintain the generator reactive power of the voltage-controlled buses within their specified limits. The violation of reactive power limit may occur if the specified voltage is either too high or too low. After a few iterations (10^{th} iteration in the Gauss method), the var calculated at the generator buses are examined. If a limit is reached, the voltage magnitude is adjusted in steps of 0.5 percent up to ± 5 percent to bring the var demand within the specified limits.

busout This program produces the bus output result in a tabulated form. The bus output result includes the voltage magnitude and angle, real and reactive power of generators and loads, and the shunt capacitor/reactor Mvar. Total generation and total load are also included as outlined in the sample case.

lineflow This program prepares the line output data. It is designed to display the active and reactive power flow entering the line terminals and line losses as well as the net power at each bus. Also included are the total real and reactive losses in the system. The output of this portion is also shown in the sample case.

6.9 DATA PREPARATION

In order to perform a power flow analysis by the Gauss-Seidel method in the *MATLAB* environment, the following variables must be defined: power system base MVA, power mismatch accuracy, acceleration factor, and maximum number of iterations. The name (in lowercase letters) reserved for these variables are **basemva, accuracy, accel**, and **maxiter**, respectively. Typical values are as follows:

```
basemva = 100;    accuracy = 0.001;
accel   = 1.6;    maxiter = 80;
```

The initial step in the preparation of input file is the numbering of each bus. Buses are numbered sequentially. Although the numbers are sequentially assigned, the buses need not be entered in sequence. In addition, the following data files are required.

BUS DATA FILE – busdata The format for the bus entry is chosen to facilitate the required data for each bus in a single row. The information required must be included in a matrix called **busdata**. Column 1 is the bus number. Column 2 contains the bus code. Columns 3 and 4 are voltage magnitude in per unit and phase angle in degrees. Columns 5 and 6 are load MW and Mvar. Column 7 through 10 are MW, Mvar, minimum Mvar and maximum Mvar of generation, in that order. The last column is the injected Mvar of shunt capacitors. The bus code entered in column 2 is used for identifying load, voltage-controlled, and slack buses as outlined below:

1 This code is used for the slack bus. The only necessary information for this bus is the voltage magnitude and its phase angle.

0 This code is used for load buses. The loads are entered positive in megawatts and megavars. For this bus, initial voltage estimate must be specified. This is usually 1 and 0 for voltage magnitude and phase angle, respectively. If voltage magnitude and phase angle for this type of bus are specified, they will be taken as the initial starting voltage for that bus instead of a flat start of 1 and 0.

2 This code is used for the voltage-controlled buses. For this bus, voltage magnitude, real power generation in megawatts, and the minimum and maximum limits of the megavar demand must be specified.

LINE DATA FILE – linedata Lines are identified by the node-pair method. The information required must be included in a matrix called **linedata**. Columns 1 and 2 are the line bus numbers. Columns 3 through 5 contain the line resistance, reactance, and one-half of the total line charging susceptance in per unit on the specified

MVA base. The last column is for the transformer tap setting; for lines, 1 must be entered in this column. The lines may be entered in any sequence or order with the only restriction being that if the entry is a transformer, the left bus number is assumed to be the tap side of the transformer.

The IEEE 30 bus system is used to demonstrate the data preparation and the use of the power flow programs by the Gauss-Seidel method.

Example 6.9

Figure 6.16 is part of the American Electric Power Service Corporation network which is being made available to the electric utility industry as a standard test case for evaluating various analytical methods and computer programs for the solution of power system problems. Use the **lfgauss** program to obtain the power solution by the Gauss-Seidel method. Bus 1 is taken as the slack bus with its voltage adjusted to $1.06\angle 0°$ pu. The data for the voltage-controlled buses is

Regulated Bus Data			
Bus No.	Voltage Magnitude	Min. Mvar Capacity	Max. Mvar Capacity
2	1.043	-40	50
5	1.010	-40	40
8	1.010	-10	40
11	1.082	-6	24
13	1.071	-6	24

Transformer tap setting are given in the table below. The left bus number is assumed to be the tap side of the transformer.

Transformer Data	
Transformer Designation	Tap Setting pu
4 – 12	0.932
6 – 9	0.978
6 – 10	0.969
28 – 27	0.968

The data for the injected Q due to shunt capacitors is

Injected Q due to Capacitors	
Bus No.	Mvar
10	19
24	4.3

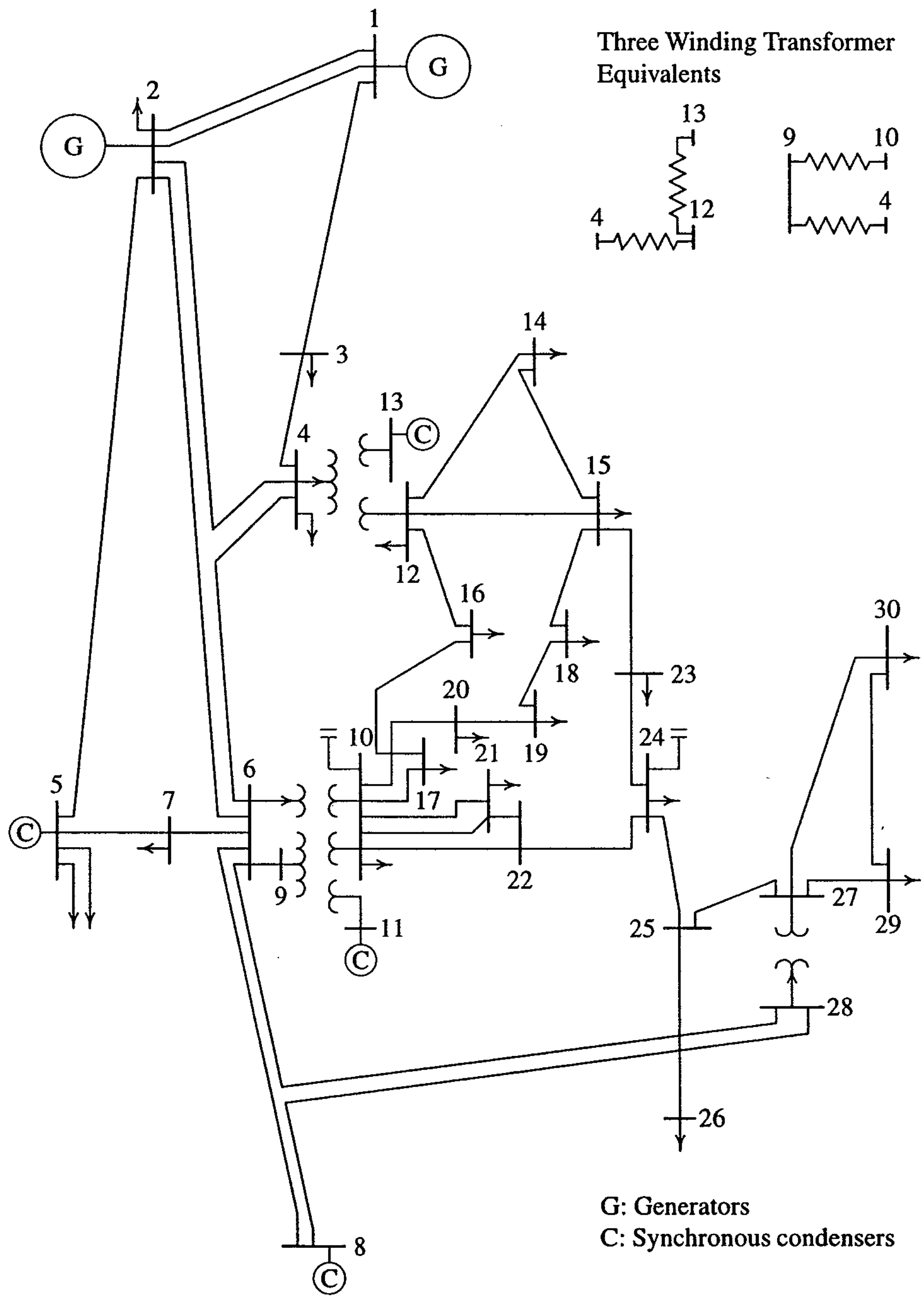

FIGURE 6.16
30-bus IEEE sample system.

Generation and loads are as given in the data prepared for use in the *MATLAB* environment in the matrix defined as **busdata**. Code 0, code 1, and code 2 are used for the load buses, the slack bus and the voltage-controlled buses, respectively. Values for **basemva, accuracy, accel** and **maxiter** must be specified. Line data are as given in the matrix called **linedata**. The last column of this data must contain 1 for lines, or the tap setting values for transformers with off-nominal turn ratio. The control commands required are **lfybus**, **lfgauss** and **lineflow**. A **diary** command may be used to save the output to the specified file name. The power flow data and the commands required are as follows.

```
clear          % clears all variables from workspace.
basemva = 100;  accuracy = 0.001; accel = 1.8; maxiter = 100;
%   IEEE 30-BUS TEST SYSTEM (American Electric Power)
%      Bus Bus Voltage Angle --Load--  ---Generator---Injected
%       No code Mag. Degree MW   Mvar   MW  Mvar Qmin Qmax Mvar
busdata=[1  1  1.06   0     0.0  0.0   0.0  0.0   0   0      0
         2  2  1.043  0    21.70 12.7  40.0  0.0 -40  50      0
         3  0  1.0    0     2.4   1.2   0.0  0.0   0   0      0
         4  0  1.06   0     7.6   1.6   0.0  0.0   0   0      0
         5  2  1.01   0    94.2  19.0   0.0  0.0 -40  40      0
         6  0  1.0    0     0.0   0.0   0.0  0.0   0   0      0
         7  0  1.0    0    22.8  10.9   0.0  0.0   0   0      0
         8  2  1.01   0    30.0  30.0   0.0  0.0 -10  40      0
         9  0  1.0    0     0.0   0.0   0.0  0.0   0   0      0
        10  0  1.0    0     5.8   2.0   0.0  0.0   0   0     19
        11  2  1.082  0     0.0   0.0   0.0  0.0  -6  24      0
        12  0  1.0    0    11.2   7.5   0    0     0   0      0
        13  2  1.071  0     0.0   0.0   0    0    -6  24      0
        14  0  1.0    0     6.2   1.6   0    0     0   0      0
        15  0  1.0    0     8.2   2.5   0    0     0   0      0
        16  0  1.0    0     3.5   1.8   0    0     0   0      0
        17  0  1.0    0     9.0   5.8   0    0     0   0      0
        18  0  1.0    0     3.2   0.9   0    0     0   0      0
        19  0  1.0    0     9.5   3.4   0    0     0   0      0
        20  0  1.0    0     2.2   0.7   0    0     0   0      0
        21  0  1.0    0    17.5  11.2   0    0     0   0      0
        22  0  1.0    0     0.0   0.0   0    0     0   0      0
        23  0  1.0    0     3.2   1.6   0    0     0   0      0
        24  0  1.0    0     8.7   6.7   0    0     0   0    4.3
        25  0  1.0    0     0.0   0.0   0    0     0   0      0
        26  0  1.0    0     3.5   2.3   0    0     0   0      0
        27  0  1.0    0     0.0   0.0   0    0     0   0      0
        28  0  1.0    0     0.0   0.0   0    0     0   0      0
        29  0  1.0    0     2.4   0.9   0    0     0   0      0
        30  0  1.0    0    10.6   1.9   0    0     0   0     0];
```

```
% Line Data
%
%         Bus   bus    R         X         1/2 B  1 for Line code or
%         nl    nr     pu        pu        pu     tap setting value
linedata=[1     2     0.0192    0.0575    0.02640     1
          1     3     0.0452    0.1852    0.02040     1
          2     4     0.0570    0.1737    0.01840     1
          3     4     0.0132    0.0379    0.00420     1
          2     5     0.0472    0.1983    0.02090     1
          2     6     0.0581    0.1763    0.01870     1
          4     6     0.0119    0.0414    0.00450     1
          5     7     0.0460    0.1160    0.01020     1
          6     7     0.0267    0.0820    0.00850     1
          6     8     0.0120    0.0420    0.00450     1
          6     9     0.0       0.2080    0.0       0.978
          6    10     0.0       0.5560    0.0       0.969
          9    11     0.0       0.2080    0.0         1
          9    10     0.0       0.1100    0.0         1
          4    12     0.0       0.2560    0.0       0.932
         12    13     0.0       0.1400    0.0         1
         12    14     0.1231    0.2559    0.0         1
         12    15     0.0662    0.1304    0.0         1
         12    16     0.0945    0.1987    0.0         1
         14    15     0.2210    0.1997    0.0         1
         16    17     0.0824    0.1923    0.0         1
         15    18     0.1073    0.2185    0.0         1
         18    19     0.0639    0.1292    0.0         1
         19    20     0.0340    0.0680    0.0         1
         10    20     0.0936    0.2090    0.0         1
         10    17     0.0324    0.0845    0.0         1
         10    21     0.0348    0.0749    0.0         1
         10    22     0.0727    0.1499    0.0         1
         21    22     0.0116    0.0236    0.0         1
         15    23     0.1000    0.2020    0.0         1
         22    24     0.1150    0.1790    0.0         1
         23    24     0.1320    0.2700    0.0         1
         24    25     0.1885    0.3292    0.0         1
         25    26     0.2544    0.3800    0.0         1
         25    27     0.1093    0.2087    0.0         1
         28    27     0.0000    0.3960    0.0       0.968
         27    29     0.2198    0.4153    0.0         1
         27    30     0.3202    0.6027    0.0         1
         29    30     0.2399    0.4533    0.0         1
          8    28     0.0636    0.2000    0.0214      1
          6    28     0.0169    0.0599    0.065       1];
```

```
%
    lfybus                    % Forms the bus admittance matrix
    lfgauss       % Power flow solution by Gauss-Seidel method
    busout       % Prints the power flow solution on the screen
    lineflow  % Computes and displays the line flow and losses
```

The **lfgauss, busout** and the **lineflow** produce the following tabulated results.

```
          Power Flow Solution by Gauss-Seidel Method
             Maximum Power mismatch = 0.000951884
                   No. of iterations = 34

Bus Voltage  Angle    -----Load-----    --Generation-- Injected
No. Mag.     Degree    MW       Mvar      MW      Mvar     Mvar

1    1.060    0.000    0.000    0.000  260.950  -17.010    0.00
2    1.043   -5.496   21.700   12.700   40.000   48.826    0.00
3    1.022   -8.002    2.400    1.200    0.000    0.000    0.00
4    1.013   -9.659    7.600    1.600    0.000    0.000    0.00
5    1.010  -14.380   94.200   19.000    0.000   35.995    0.00
6    1.012  -11.396    0.000    0.000    0.000    0.000    0.00
7    1.003  -13.149   22.800   10.900    0.000    0.000    0.00
8    1.010  -12.114   30.000   30.000    0.000   30.759    0.00
9    1.051  -14.432    0.000    0.000    0.000    0.000    0.00
10   1.044  -16.024    5.800    2.000    0.000    0.000   19.00
11   1.082  -14.432    0.000    0.000    0.000   16.113    0.00
12   1.057  -15.301   11.200    7.500    0.000    0.000    0.00
13   1.071  -15.300    0.000    0.000    0.000   10.406    0.00
14   1.043  -16.190    6.200    1.600    0.000    0.000    0.00
15   1.038  -16.276    8.200    2.500    0.000    0.000    0.00
16   1.045  -15.879    3.500    1.800    0.000    0.000    0.00
17   1.039  -16.187    9.000    5.800    0.000    0.000    0.00
18   1.028  -16.881    3.200    0.900    0.000    0.000    0.00
19   1.025  -17.049    9.500    3.400    0.000    0.000    0.00
20   1.029  -16.851    2.200    0.700    0.000    0.000    0.00
21   1.032  -16.468   17.500   11.200    0.000    0.000    0.00
22   1.033  -16.455    0.000    0.000    0.000    0.000    0.00
23   1.027  -16.660    3.200    1.600    0.000    0.000    0.00
24   1.022  -16.829    8.700    6.700    0.000    0.000    4.30
25   1.019  -16.423    0.000    0.000    0.000    0.000    0.00
26   1.001  -16.835    3.500    2.300    0.000    0.000    0.00
27   1.026  -15.913    0.000    0.000    0.000    0.000    0.00
28   1.011  -12.056    0.000    0.000    0.000    0.000    0.00
```

29	1.006	-17.133	2.400	0.900	0.000	0.000	0.00
30	0.994	-18.016	10.600	1.900	0.000	0.000	0.00
Total			283.400	126.200	300.950	125.089	23.30

Line Flow and Losses

--Line--		Power at bus & line flow			--Line loss--		Transformer
from	to	MW	Mvar	MVA	MW	Mvar	tap
1		260.950	-17.010	261.504			
	2	177.743	-22.140	179.117	5.461	10.517	
	3	83.197	5.125	83.354	2.807	7.079	
2		18.300	36.126	40.497			
	1	-172.282	32.657	175.350	5.461	10.517	
	4	45.702	2.720	45.783	1.106	-0.519	
	5	82.990	1.704	83.008	2.995	8.178	
	6	61.905	-0.966	61.913	2.047	2.263	
3		-2.400	-1.200	2.683			
	1	-80.390	1.954	80.414	2.807	7.079	
	4	78.034	-3.087	78.095	0.771	1.345	
4		-7.600	-1.600	7.767			
	2	-44.596	-3.239	44.713	1.106	-0.519	
	3	-77.263	4.432	77.390	0.771	1.345	
	6	70.132	-17.624	72.313	0.605	1.181	
	12	44.131	14.627	46.492	0.000	4.686	0.932
5		-94.200	16.995	95.721			
	2	-79.995	6.474	80.256	2.995	8.178	
	7	-14.210	10.467	17.649	0.151	-1.687	
6		0.000	0.000	0.000			
	2	-59.858	3.229	59.945	2.047	2.263	
	4	-69.527	18.805	72.026	0.605	1.181	
	7	37.537	-1.915	37.586	0.368	-0.598	
	8	29.534	-3.712	29.766	0.103	-0.558	
	9	27.687	-7.318	28.638	0.000	1.593	0.978
	10	15.828	0.656	15.842	0.000	1.279	0.969
	28	18.840	-9.575	21.134	0.060	-13.085	
7		-22.800	-10.900	25.272			
	5	14.361	-12.154	18.814	0.151	-1.687	

	6	-37.170	1.317	37.193	0.368	-0.598
8		-30.000	0.759	30.010		
	6	-29.431	3.154	29.599	0.103	-0.558
	28	-0.570	-2.366	2.433	0.000	-4.368
9		0.000	0.000	0.000		
	6	-27.687	8.911	29.086	0.000	1.593
	11	0.003	-15.653	15.653	-0.000	0.461
	10	27.731	6.747	28.540	0.000	0.811
10		-5.800	17.000	17.962		
	6	-15.828	0.623	15.840	0.000	1.279
	9	-27.731	-5.936	28.359	0.000	0.811
	20	9.018	3.569	9.698	0.081	0.180
	17	5.347	4.393	6.920	0.014	0.037
	21	15.723	9.846	18.551	0.110	0.236
	22	7.582	4.487	8.811	0.052	0.107
11		0.000	16.113	16.113		
	9	-0.003	16.114	16.114	-0.000	0.461
12		-11.200	-7.500	13.479		
	4	-44.131	-9.941	45.237	0.000	4.686
	13	-0.021	-10.274	10.274	0.000	0.132
	14	7.852	2.428	8.219	0.074	0.155
	15	17.852	6.968	19.164	0.217	0.428
	16	7.206	3.370	7.955	0.053	0.112
13		0.000	10.406	10.406		
	12	0.021	10.406	10.406	0.000	0.132
14		-6.200	-1.600	6.403		
	12	-7.778	-2.273	8.103	0.074	0.155
	15	1.592	0.708	1.742	0.006	0.006
15		-8.200	-2.500	8.573		
	12	-17.634	-6.540	18.808	0.217	0.428
	14	-1.586	-0.702	1.734	0.006	0.006
	18	6.009	1.741	6.256	0.039	0.079
	23	5.004	2.963	5.815	0.031	0.063
16		-3.500	-1.800	3.936		
	12	-7.152	-3.257	7.859	0.053	0.112
	17	3.658	1.440	3.931	0.012	0.027

17		-9.000	-5.800	10.707		
	16	-3.646	-1.413	3.910	0.012	0.027
	10	-5.332	-4.355	6.885	0.014	0.037
18		-3.200	-0.900	3.324		
	15	-5.970	-1.661	6.197	0.039	0.079
	19	2.779	0.787	2.888	0.005	0.010
19		-9.500	-3.400	10.090		
	18	-2.774	-0.777	2.881	0.005	0.010
	20	-6.703	-2.675	7.217	0.017	0.034
20		-2.200	-0.700	2.309		
	19	6.720	2.709	7.245	0.017	0.034
	10	-8.937	-3.389	9.558	0.081	0.180
21		-17.500	-11.200	20.777		
	10	-15.613	-9.609	18.333	0.110	0.236
	22	-1.849	-1.627	2.463	0.001	0.001
22		0.000	0.000	0.000		
	10	-7.531	-4.380	8.712	0.052	0.107
	21	1.850	1.628	2.464	0.001	0.001
	24	5.643	2.795	6.297	0.043	0.067
23		-3.200	-1.600	3.578		
	15	-4.972	-2.900	5.756	0.031	0.063
	24	1.771	1.282	2.186	0.006	0.012
24		-8.700	-2.400	9.025		
	22	-5.601	-2.728	6.230	0.043	0.067
	23	-1.765	-1.270	2.174	0.006	0.012
	25	-1.322	1.604	2.079	0.008	0.014
25		0.000	0.000	0.000		
	24	1.330	-1.590	2.073	0.008	0.014
	26	3.520	2.372	4.244	0.044	0.066
	27	-4.866	-0.786	4.929	0.026	0.049
26		-3.500	-2.300	4.188		
	25	-3.476	-2.306	4.171	0.044	0.066
27		0.000	0.000	0.000		
	25	4.892	0.835	4.963	0.026	0.049

	28	-18.192	-4.152	18.660	-0.000	1.310	
	29	6.178	1.675	6.401	0.086	0.162	
	30	7.093	1.663	7.286	0.162	0.304	
28		0.000	0.000	0.000			
	27	18.192	5.463	18.994	-0.000	1.310	0.968
	8	0.570	-2.003	2.082	0.000	-4.368	
	6	-18.780	-3.510	19.106	0.060	-13.085	
29		-2.400	-0.900	2.563			
	27	-6.093	-1.513	6.278	0.086	0.162	
	30	3.716	0.601	3.764	0.034	0.063	
30		-10.600	-1.900	10.769			
	27	-6.932	-1.359	7.064	0.162	0.304	
	29	-3.683	-0.537	3.722	0.034	0.063	
Total loss					17.594	22.233	

6.10 NEWTON-RAPHSON POWER FLOW SOLUTION

Because of its quadratic convergence, Newton's method is mathematically superior to the Gauss-Seidel method and is less prone to divergence with ill-conditioned problems. For large power systems, the Newton-Raphson method is found to be more efficient and practical. The number of iterations required to obtain a solution is independent of the system size, but more functional evaluations are required at each iteration. Since in the power flow problem real power and voltage magnitude are specified for the voltage-controlled buses, the power flow equation is formulated in polar form. For the typical bus of the power system shown in Figure 6.7, the current entering bus i is given by (6.24). This equation can be rewritten in terms of the bus admittance matrix as

$$I_i = \sum_{j=1}^{n} Y_{ij} V_j \tag{6.48}$$

In the above equation, j includes bus i. Expressing this equation in polar form, we have

$$I_i = \sum_{j=1}^{n} |Y_{ij}||V_j| \angle \theta_{ij} + \delta_j \tag{6.49}$$

The complex power at bus i is

$$P_i - jQ_i = V_i^* I_i \tag{6.50}$$

Substituting from (6.49) for I_i in (6.50),

$$P_i - jQ_i = |V_i|\angle{-\delta_i}\sum_{j=1}^{n}|Y_{ij}||V_j|\angle\theta_{ij}+\delta_j \tag{6.51}$$

Separating the real and imaginary parts,

$$P_i = \sum_{j=1}^{n}|V_i||V_j||Y_{ij}|\cos(\theta_{ij}-\delta_i+\delta_j) \tag{6.52}$$

$$Q_i = -\sum_{j=1}^{n}|V_i||V_j||Y_{ij}|\sin(\theta_{ij}-\delta_i+\delta_j) \tag{6.53}$$

Equations (6.52) and (6.53) constitute a set of nonlinear algebraic equations in terms of the independent variables, voltage magnitude in per unit, and phase angle in radians. We have two equations for each load bus, given by (6.52) and (6.53), and one equation for each voltage-controlled bus, given by (6.52). Expanding (6.52) and (6.53) in Taylor's series about the initial estimate and neglecting all higher order terms results in the following set of linear equations.

$$\left[\begin{array}{c} \Delta P_2^{(k)} \\ \vdots \\ \Delta P_n^{(k)} \\ \hline \Delta Q_2^{(k)} \\ \vdots \\ \Delta Q_n^{(k)} \end{array}\right] = \left[\begin{array}{ccc|ccc} \frac{\partial P_2}{\partial \delta_2}^{(k)} & \cdots & \frac{\partial P_2}{\partial \delta_n}^{(k)} & \frac{\partial P_2}{\partial |V_2|}^{(k)} & \cdots & \frac{\partial P_2}{\partial |V_n|}^{(k)} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial P_n}{\partial \delta_2}^{(k)} & \cdots & \frac{\partial P_n}{\partial \delta_n}^{(k)} & \frac{\partial P_n}{\partial |V_2|}^{(k)} & \cdots & \frac{\partial P_n}{\partial |V_n|}^{(k)} \\ \hline \frac{\partial Q_2}{\partial \delta_2}^{(k)} & \cdots & \frac{\partial Q_2}{\partial \delta_n}^{(k)} & \frac{\partial Q_2}{\partial |V_2|}^{(k)} & \cdots & \frac{\partial Q_2}{\partial |V_n|}^{(k)} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial Q_n}{\partial \delta_2}^{(k)} & \cdots & \frac{\partial Q_n}{\partial \delta_n}^{(k)} & \frac{\partial Q_n}{\partial |V_2|}^{(k)} & \cdots & \frac{\partial Q_n}{\partial |V_n|}^{(k)} \end{array}\right] \left[\begin{array}{c} \Delta \delta_2^{(k)} \\ \vdots \\ \Delta \delta_n^{(k)} \\ \hline \Delta |V_2^{(k)}| \\ \vdots \\ \Delta |V_n^{(k)}| \end{array}\right]$$

In the above equation, bus 1 is assumed to be the slack bus. The Jacobian matrix gives the linearized relationship between small changes in voltage angle $\Delta\delta_i^{(k)}$ and voltage magnitude $\Delta|V_i^{(k)}|$ with the small changes in real and reactive power $\Delta P_i^{(k)}$ and $\Delta Q_i^{(k)}$. Elements of the Jacobian matrix are the partial derivatives of (6.52) and (6.53), evaluated at $\Delta\delta_i^{(k)}$ and $\Delta|V_i^{(k)}|$. In short form, it can be written as

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \begin{bmatrix} \Delta\delta \\ \Delta|V| \end{bmatrix} \tag{6.54}$$

For voltage-controlled buses, the voltage magnitudes are known. Therefore, if m buses of the system are voltage-controlled, m equations involving ΔQ and ΔV

and the corresponding columns of the Jacobian matrix are eliminated. Accordingly, there are $n-1$ real power constraints and $n-1-m$ reactive power constraints, and the Jacobian matrix is of order $(2n-2-m) \times (2n-2-m)$. $\boldsymbol{J_1}$ is of the order $(n-1) \times (n-1)$, $\boldsymbol{J_2}$ is of the order $(n-1) \times (n-1-m)$, $\boldsymbol{J_3}$ is of the order $(n-1-m) \times (n-1)$, and $\boldsymbol{J_4}$ is of the order $(n-1-m) \times (n-1-m)$.

The diagonal and the off-diagonal elements of $\boldsymbol{J_1}$ are

$$\frac{\partial P_i}{\partial \delta_i} = \sum_{j \neq i} |V_i||V_j||Y_{ij}| \sin(\theta_{ij} - \delta_i + \delta_j) \tag{6.55}$$

$$\frac{\partial P_i}{\partial \delta_j} = -|V_i||V_j||Y_{ij}| \sin(\theta_{ij} - \delta_i + \delta_j) \qquad j \neq i \tag{6.56}$$

The diagonal and the off-diagonal elements of $\boldsymbol{J_2}$ are

$$\frac{\partial P_i}{\partial |V_i|} = 2|V_i||Y_{ii}| \cos\theta_{ii} + \sum_{j \neq i} |V_j||Y_{ij}| \cos(\theta_{ij} - \delta_i + \delta_j) \tag{6.57}$$

$$\frac{\partial P_i}{\partial |V_j|} = |V_i||Y_{ij}| \cos(\theta_{ij} - \delta_i + \delta_j) \qquad j \neq i \tag{6.58}$$

The diagonal and the off-diagonal elements of $\boldsymbol{J_3}$ are

$$\frac{\partial Q_i}{\partial \delta_i} = \sum_{j \neq i} |V_i||V_j||Y_{ij}| \cos(\theta_{ij} - \delta_i + \delta_j) \tag{6.59}$$

$$\frac{\partial Q_i}{\partial \delta_j} = -|V_i||V_j||Y_{ij}| \cos(\theta_{ij} - \delta_i + \delta_j) \qquad j \neq i \tag{6.60}$$

The diagonal and the off-diagonal elements of $\boldsymbol{J_4}$ are

$$\frac{\partial Q_i}{\partial |V_i|} = -2|V_i||Y_{ii}| \sin\theta_{ii} - \sum_{j \neq i} |V_j||Y_{ij}| \sin(\theta_{ij} - \delta_i + \delta_j) \tag{6.61}$$

$$\frac{\partial Q_i}{\partial |V_j|} = -|V_i||Y_{ij}| \sin(\theta_{ij} - \delta_i + \delta_j) \qquad j \neq i \tag{6.62}$$

The terms $\Delta P_i^{(k)}$ and $\Delta Q_i^{(k)}$ are the difference between the scheduled and calculated values, known as the *power residuals*, given by

$$\Delta P_i^{(k)} = P_i^{sch} - P_i^{(k)} \tag{6.63}$$

$$\Delta Q_i^{(k)} = Q_i^{sch} - Q_i^{(k)} \tag{6.64}$$

The new estimates for bus voltages are

$$\delta_i^{(k+1)} = \delta_i^{(k)} + \Delta\delta_i^{(k)} \tag{6.65}$$

$$|V_i^{(k+1)}| = |V_i^{(k)}| + \Delta|V_i^{(k)}| \tag{6.66}$$

The procedure for power flow solution by the Newton-Raphson method is as follows:

1. For load buses, where P_i^{sch} and Q_i^{sch} are specified, voltage magnitudes and phase angles are set equal to the slack bus values, or 1.0 and 0.0, i.e., $|V_i^{(0)}| = 1.0$ and $\delta_i^{(0)} = 0.0$. For voltage-regulated buses, where $|V_i|$ and P_i^{sch} are specified, phase angles are set equal to the slack bus angle, or 0, i.e., $\delta_i^{(0)} = 0$.
2. For load buses, $P_i^{(k)}$ and $Q_i^{(k)}$ are calculated from (6.52) and (6.53) and $\Delta P_i^{(k)}$ and $\Delta Q_i^{(k)}$ are calculated from (6.63) and (6.64).
3. For voltage-controlled buses, $P_i^{(k)}$ and $\Delta P_i^{(k)}$ are calculated from (6.52) and (6.63), respectively.
4. The elements of the Jacobian matrix (J_1, J_2, J_3, and J_4) are calculated from (6.55) – (6.62).
5. The linear simultaneous equation (6.54) is solved directly by optimally ordered triangular factorization and Gaussian elimination.
6. The new voltage magnitudes and phase angles are computed from (6.65) and (6.66).
7. The process is continued until the residuals $\Delta P_i^{(k)}$ and $\Delta Q_i^{(k)}$ are less than the specified accuracy, i.e.,

$$\begin{aligned} |\Delta P_i^{(k)}| &\le \epsilon \\ |\Delta Q_i^{(k)}| &\le \epsilon \end{aligned} \tag{6.67}$$

The power flow solution by the Newton-Raphson method is demonstrated in the following example.

Example 6.10

Obtain the power flow solution by the Newton-Raphson method for the system of Example 6.8.

Line impedances converted to admittances are $y_{12} = 10 - j20$, $y_{13} = 10 - j30$, and $y_{23} = 16 - j32$. This results in the bus admittance matrix

$$Y_{bus} = \begin{bmatrix} 20 - j50 & -10 + j20 & -10 + j30 \\ -10 + j20 & 26 - j52 & -16 + j32 \\ -10 + j30 & -16 + j32 & 26 - j62 \end{bmatrix}$$

Converting the bus admittance matrix to polar form with angles in radian yields

$$Y_{bus} = \begin{bmatrix} 53.85165\angle -1.9029 & 22.36068\angle 2.0344 & 31.62278\angle 1.8925 \\ 22.36068\angle 2.0344 & 58.13777\angle -1.1071 & 35.77709\angle 2.0344 \\ 31.62278\angle 1.8925 & 35.77709\angle 2.0344 & 67.23095\angle -1.1737 \end{bmatrix}$$

From (6.52) and (6.53), the expressions for real power at bus 2 and 3 and the reactive power at bus 2 are

$$\begin{aligned} P_2 &= |V_2||V_1||Y_{21}|\cos(\theta_{21} - \delta_2 + \delta_1) + |V_2^2||Y_{22}|\cos\theta_{22} + \\ &\quad |V_2||V_3||Y_{23}|\cos(\theta_{23} - \delta_2 + \delta_3) \\ P_3 &= |V_3||V_1||Y_{31}|\cos(\theta_{31} - \delta_3 + \delta_1) + |V_3||V_2||Y_{32}|\cos(\theta_{32} - \\ &\quad \delta_3 + \delta_2) + |V_3^2||Y_{33}|\cos\theta_{33} \\ Q_2 &= -|V_2||V_1||Y_{21}|\sin(\theta_{21} - \delta_2 + \delta_1) - |V_2^2||Y_{22}|\sin\theta_{22} - \\ &\quad |V_2||V_3||Y_{23}|\sin(\theta_{23} - \delta_2 + \delta_3) \end{aligned}$$

Elements of the Jacobian matrix are obtained by taking partial derivatives of the above equations with respect to δ_2, δ_3 and $|V_2|$.

$$\begin{aligned} \frac{\partial P_2}{\partial \delta_2} &= |V_2||V_1||Y_{21}|\sin(\theta_{21} - \delta_2 + \delta_1) + |V_2||V_3||Y_{23}| \\ &\quad \sin(\theta_{23} - \delta_2 + \delta_3) \\ \frac{\partial P_2}{\partial \delta_3} &= -|V_2||V_3||Y_{23}|\sin(\theta_{23} - \delta_2 + \delta_3) \\ \frac{\partial P_2}{\partial |V_2|} &= |V_1||Y_{21}|\cos(\theta_{21} - \delta_2 + \delta_1) + 2|V_2||Y_{22}|\cos\theta_{22} + \\ &\quad |V_3||Y_{23}|\cos(\theta_{23} - \delta_2 + \delta_3) \\ \frac{\partial P_3}{\partial \delta_2} &= -|V_3||V_2||Y_{32}|\sin(\theta_{32} - \delta_3 + \delta_2) \\ \frac{\partial P_3}{\partial \delta_3} &= |V_3||V_1||Y_{31}|\sin(\theta_{31} - \delta_3 + \delta_1) + |V_3||V_2||Y_{32}| \\ &\quad \sin(\theta_{32} - \delta_3 + \delta_2) \\ \frac{\partial P_3}{\partial |V_2|} &= |V_3||Y_{32}|\cos(\theta_{32} - \delta_3 + \delta_2) \end{aligned}$$

$$\frac{\partial Q_2}{\partial \delta_2} = |V_2||V_1||Y_{21}|\cos(\theta_{21} - \delta_2 + \delta_1) + |V_2||V_3||Y_{23}| \cos(\theta_{23} - \delta_2 + \delta_3)$$

$$\frac{\partial Q_2}{\partial \delta_3} = -|V_2||V_3||Y_{23}|\cos(\theta_{23} - \delta_2 + \delta_3)$$

$$\frac{\partial Q_2}{\partial |V_2|} = -|V_1||Y_{21}|\sin(\theta_{21} - \delta_2 + \delta_1) - 2|V_2||Y_{22}|\sin\theta_{22} - |V_3||Y_{23}|\sin(\theta_{23} - \delta_2 + \delta_3)$$

The load and generation expressed in per units are

$$S_2^{sch} = -\frac{(400 + j250)}{100} = -4.0 - j2.5 \quad \text{pu}$$

$$P_3^{sch} = \frac{200}{100} = 2.0 \quad \text{pu}$$

The slack bus voltage is $V_1 = 1.05\angle 0$ pu, and the bus 3 voltage magnitude is $|V_3| = 1.04$ pu. Starting with an initial estimate of $|V_2^{(0)}| = 1.0$, $\delta_2^{(0)} = 0.0$, and $\delta_3^{(0)} = 0.0$, the power residuals are computed from (6.63) and (6.64)

$$\Delta P_2^{(0)} = P_2^{sch} - P_2^{(0)} = -4.0 - (-1.14) = -2.8600$$
$$\Delta P_3^{(0)} = P_3^{sch} - P_3^{(0)} = 2.0 - (0.5616) = 1.4384$$
$$\Delta Q_2^{(0)} = Q_2^{sch} - Q_2^{(0)} = -2.5 - (-2.28) = -0.2200$$

Evaluating the elements of the Jacobian matrix with the initial estimate, the set of linear equations in the first iteration becomes

$$\begin{bmatrix} -2.8600 \\ 1.4384 \\ -0.2200 \end{bmatrix} = \begin{bmatrix} 54.28000 & -33.28000 & 24.86000 \\ -33.28000 & 66.04000 & -16.64000 \\ -27.14000 & 16.64000 & 49.72000 \end{bmatrix} \begin{bmatrix} \Delta\delta_2^{(0)} \\ \Delta\delta_3^{(0)} \\ \Delta|V_2^{(0)}| \end{bmatrix}$$

Obtaining the solution of the above matrix equation, the new bus voltages in the first iteration are

$$\Delta\delta_2^{(0)} = -0.045263 \qquad \delta_2^{(1)} = 0 + (-0.045263) = -0.045263$$
$$\Delta\delta_3^{(0)} = -0.007718 \qquad \delta_3^{(1)} = 0 + (-0.007718) = -0.007718$$
$$\Delta|V_2^{(0)}| = -0.026548 \qquad |V_2^{(1)}| = 1 + (-0.026548) = 0.97345$$

Voltage phase angles are in radians. For the second iteration, we have

$$\begin{bmatrix} -0.099218 \\ 0.021715 \\ -0.050914 \end{bmatrix} = \begin{bmatrix} 51.724675 & -31.765618 & 21.302567 \\ -32.981642 & 65.656383 & -15.379086 \\ -28.538577 & 17.402838 & 48.103589 \end{bmatrix} \begin{bmatrix} \Delta\delta_2^{(1)} \\ \Delta\delta_3^{(1)} \\ \Delta|V_2^{(1)}| \end{bmatrix}$$

and

$$\begin{aligned}
\Delta\delta_2^{(1)} &= -0.001795 & \delta_2^{(2)} &= -0.045263 + (-0.001795) = -0.04706 \\
\Delta\delta_3^{(1)} &= -0.000985 & \delta_3^{(2)} &= -0.007718 + (-0.000985) = -0.00870 \\
\Delta|V_2^{(1)}| &= -0.001767 & |V_2^{(2)}| &= 0.973451 + (-0.001767) = 0.971684
\end{aligned}$$

For the third iteration, we have

$$\begin{bmatrix} -0.000216 \\ 0.000038 \\ -0.000143 \end{bmatrix} = \begin{bmatrix} 51.596701 & -31.693866 & 21.147447 \\ -32.933865 & 65.597585 & -15.351628 \\ -28.548205 & 17.396932 & 47.954870 \end{bmatrix} \begin{bmatrix} \Delta\delta_2^{(2)} \\ \Delta\delta_3^{(2)} \\ \Delta|V_2^{(2)}| \end{bmatrix}$$

and

$$\begin{aligned}
\Delta\delta_2^{(2)} &= -0.000038 & \delta_2^{(3)} &= -0.047058 + (-0.0000038) = -0.04706 \\
\Delta\delta_3^{(2)} &= -0.0000024 & \delta_3^{(3)} &= -0.008703 + (-0.0000024) = \ 0.008705 \\
\Delta|V_2^{(2)}| &= -0.0000044 & |V_2^{(3)}| &= 0.971684 + (-0.0000044) = 0.97168
\end{aligned}$$

The solution converges in 3 iterations with a maximum power mismatch of 2.5×10^{-4} with $V_2 = 0.97168\angle{-2.696°}$ and $V_3 = 1.04\angle{-0.4988°}$. From (6.52) and (6.53), the expressions for reactive power at bus 3 and the slack bus real and reactive powers are

$$\begin{aligned}
Q_3 &= -|V_3||V_1||Y_{31}|\sin(\theta_{31} - \delta_3 + \delta_1) - |V_3||V_2||Y_{32}| \\
&\quad \sin(\theta_{32} - \delta_3 + \delta_2) - |V_3|^2|Y_{33}|\sin\theta_{33} \\
P_1 &= |V_1|^2|Y_{11}|\cos\theta_{11} + |V_1||V_2||Y_{12}|\cos(\theta_{12} - \delta_1 + \delta_2) + |V_1||V_3| \\
&\quad |Y_{13}|\cos(\theta_{13} - \delta_1 + \delta_3) \\
Q_1 &= -|V_1|^2|Y_{11}|\sin\theta_{11} - |V_1||V_2||Y_{12}|\sin(\theta_{12} - \delta_1 + \delta_2) - |V_1||V_3| \\
&\quad |Y_{13}|\sin(\theta_{13} - \delta_1 + \delta_3)
\end{aligned}$$

Upon substitution, we have

$$\begin{aligned}
Q_3 &= 1.4617 \text{ pu} \\
P_1 &= 2.1842 \text{ pu} \\
Q_1 &= 1.4085 \text{ pu}
\end{aligned}$$

Finally, the line flows are calculated in the same manner as the line flow calculations in the Gauss-Seidel method described in Example 6.7, and the power flow diagram is as shown in Figure 6.13.

A program named **lfnewton** is developed for power flow solution by the Newton-Raphson method for practical power systems. This program must be preceded by the **lfybus** program. **busout** and **lineflow** programs can be used to print the load flow solution and the line flow results. The format is the same as the Gauss-Seidel. The following is a brief description of the **lfnewton** program.

lfnewton This program obtains the power flow solution by the Newton-Raphson method and requires the **busdata** and the **linedata** files described in Section 6.9. It is designed for the direct use of load and generation in MW and Mvar, bus voltages in per unit, and angle in degrees. Loads and generation are converted to per unit quantities on the base MVA selected. A provision is made to maintain the generator reactive power of the voltage-controlled buses within their specified limits. The violation of reactive power limit may occur if the specified voltage is either too high or too low. In the second iteration, the var calculated at the generator buses are examined. If a limit is reached, the voltage magnitude is adjusted in steps of 0.5 percent up to ±5 percent to bring the var demand within the specified limits.

Example 6.11

Obtain the power flow solution for the IEEE-30 bus test system by the Newton-Raphson method.

The data required is the same as in Example 6.9 with the following commands

```
clear       % clears all variables from the workspace.
basemva = 100;  accuracy = 0.001;  maxiter = 12;

busdata =  [ same as in Example 6.9 ];
linedata = [ same as in Example 6.9 ];

    lfybus                   % Forms the bus admittance matrix
    lfnewton   % Power flow solution by Newton-Raphson method
    busout     % Prints the power flow solution on the screen
    lineflow % Computes and displays the line flow and losses
```

The output of **lfnewton** is

```
          Power Flow Solution by Newton-Raphson Method
             Maximum Power mismatch = 7.54898e-07
                    No. of iterations = 4

Bus Voltage  Angle    -----Load-----    --Generation-- Injected
No.  Mag.    Degree    MW      Mvar      MW       Mvar     Mvar

1   1.060    0.000    0.000    0.000  260.998  -17.021     0.00
2   1.043   -5.497   21.700   12.700   40.000   48.822     0.00
3   1.022   -8.004    2.400    1.200    0.000    0.000     0.00
```

4	1.013	-9.661	7.600	1.600	0.000	0.000	0.00
5	1.010	-14.381	94.200	19.000	0.000	35.975	0.00
6	1.012	-11.398	0.000	0.000	0.000	0.000	0.00
7	1.003	-13.150	22.800	10.900	0.000	0.000	0.00
8	1.010	-12.115	30.000	30.000	0.000	30.826	0.00
9	1.051	-14.434	0.000	0.000	0.000	0.000	0.00
10	1.044	-16.024	5.800	2.000	0.000	0.000	19.00
11	1.082	-14.434	0.000	0.000	0.000	16.119	0.00
12	1.057	-15.302	11.200	7.500	0.000	0.000	0.00
13	1.071	-15.302	0.000	0.000	0.000	10.423	0.00
14	1.042	-16.191	6.200	1.600	0.000	0.000	0.00
15	1.038	-16.278	8.200	2.500	0.000	0.000	0.00
16	1.045	-15.880	3.500	1.800	0.000	0.000	0.00
17	1.039	-16.188	9.000	5.800	0.000	0.000	0.00
18	1.028	-16.884	3.200	0.900	0.000	0.000	0.00
19	1.025	-17.052	9.500	3.400	0.000	0.000	0.00
20	1.029	-16.852	2.200	0.700	0.000	0.000	0.00
21	1.032	-16.468	17.500	11.200	0.000	0.000	0.00
22	1.033	-16.455	0.000	0.000	0.000	0.000	0.00
23	1.027	-16.662	3.200	1.600	0.000	0.000	0.00
24	1.022	-16.830	8.700	6.700	0.000	0.000	4.30
25	1.019	-16.424	0.000	0.000	0.000	0.000	0.00
26	1.001	-16.842	3.500	2.300	0.000	0.000	0.00
27	1.026	-15.912	0.000	0.000	0.000	0.000	0.00
28	1.011	-12.057	0.000	0.000	0.000	0.000	0.00
29	1.006	-17.136	2.400	0.900	0.000	0.000	0.00
30	0.995	-18.015	10.600	1.900	0.000	0.000	0.00
Total			283.400	126.200	300.998	125.144	23.30

The output of the **lineflow** is the same as the line flow output of Example 6.9 with the power mismatch as dictated by the Newton-Raphson method.

6.11 FAST DECOUPLED POWER FLOW SOLUTION

Power system transmission lines have a very high X/R ratio. For such a system, real power changes ΔP are less sensitive to changes in the voltage magnitude and are most sensitive to changes in phase angle $\Delta\delta$. Similarly, reactive power is less sensitive to changes in angle and are mainly dependent on changes in voltage magnitude. Therefore, it is reasonable to set elements J_2 and J_3 of the Jacobian matrix to zero. Thus, (6.54) becomes

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} J_1 & 0 \\ 0 & J_4 \end{bmatrix} \begin{bmatrix} \Delta\delta \\ \Delta|V| \end{bmatrix} \tag{6.68}$$

or

$$\Delta P = J_1 \Delta\delta = [\frac{\partial P}{\partial \delta}]\ \Delta\delta \tag{6.69}$$

$$\Delta Q = J_4 \Delta|V| = [\frac{\partial Q}{\partial |V|}]\ \Delta|V| \tag{6.70}$$

(6.69) and (6.70) show that the matrix equation is separated into two decoupled equations requiring considerably less time to solve compared to the time required for the solution of (6.54). Furthermore, considerable simplification can be made to eliminate the need for recomputing J_1 and J_4 during each iteration. This procedure results in the decoupled power flow equations developed by Stott and Alsac[75–76]. The diagonal elements of J_1 described by (6.55) may be written as

$$\frac{\partial P_i}{\partial \delta_i} = \sum_{j=1}^{n} |V_i||V_j||Y_{ij}|\sin(\theta_{ij} - \delta_i + \delta_j) - |V_i|^2|Y_{ii}|\sin\theta_{ii}$$

Replacing the first term of the above equation with $-Q_i$, as given by (6.53), results in

$$\begin{aligned}\frac{\partial P_i}{\partial \delta_i} &= -Q_i - |V_i|^2|Y_{ii}|\sin\theta_{ii} \\ &= -Q_i - |V_i|^2 B_{ii}\end{aligned}$$

Where $B_{ii} = |Y_{ii}|\sin\theta_{ii}$ is the imaginary part of the diagonal elements of the bus admittance matrix. B_{ii} is the sum of susceptances of all the elements incident to bus i. In a typical power system, the self-susceptance $B_{ii} \gg Q_i$, and we may neglect Q_i. Further simplification is obtained by assuming $|V_i|^2 \approx |V_i|$, which yields

$$\frac{\partial P_i}{\partial \delta_i} = -|V_i|B_{ii} \tag{6.71}$$

Under normal operating conditions, $\delta_j - \delta_i$ is quite small. Thus, in (6.56) assuming $\theta_{ii} - \delta_i + \delta_j \approx \theta_{ii}$, the off-diagonal elements of J_1 becomes

$$\frac{\partial P_i}{\partial \delta_j} = -|V_i||V_j|B_{ij}$$

Further simplification is obtained by assuming $|V_j| \approx 1$

$$\frac{\partial P_i}{\partial \delta_j} = -|V_i|B_{ij} \tag{6.72}$$

Similarly, the diagonal elements of J_4 described by (6.61) may be written as

$$\frac{\partial Q_i}{\partial |V_i|} = -|V_i||Y_{ii}|\sin\theta_{ii} - \sum_{j=1}^{n} |V_i||V_j||Y_{ij}|\sin(\theta_{ij} - \delta_i + \delta_j)$$

replacing the second term of the above equation with $-Q_i$, as given by (6.53), results in

$$\frac{\partial Q_i}{\partial |V_i|} = -|V_i||Y_{ii}|\sin\theta_{ii} + Q_i$$

Again, since $B_{ii} = Y_{ii}\sin\theta_{ii} \gg Q_i$, Q_i may be neglected and (6.61) reduces to

$$\frac{\partial Q_i}{\partial |V_i|} = -|V_i|B_{ii} \tag{6.73}$$

Likewise in (6.62), assuming $\theta_{ij} - \delta_i + \delta_j \approx \theta_{ij}$ yields

$$\frac{\partial Q_i}{\partial |V_j|} = -|V_i|B_{ij} \tag{6.74}$$

With these assumptions, equations (6.69) and (6.70) take the following form

$$\frac{\Delta P}{|V_i|} = -B'\ \Delta\delta \tag{6.75}$$

$$\frac{\Delta Q}{|V_i|} = -B''\ \Delta|V| \tag{6.76}$$

Here, B' and B'' are the imaginary part of the bus admittance matrix Y_{bus}. Since the elements of this matrix are constant, they need to be triangularized and inverted only once at the beginning of the iteration. B' is of order of $(n-1)$. For voltage-controlled buses where $|V_i|$ and P_i are specified and Q_i is not specified, the corresponding row and column of Y_{bus} are eliminated. Thus, B'' is of order of $(n-1-m)$, where m is the number of voltage-regulated buses. Therefore, in the fast decoupled power flow algorithm, the successive voltage magnitude and phase angle changes are

$$\Delta\delta = -[B']^{-1}\frac{\Delta P}{|V|} \tag{6.77}$$

$$\Delta|V| = -[B'']^{-1}\frac{\Delta Q}{|V|} \tag{6.78}$$

The fast decoupled power flow solution requires more iterations than the Newton-Raphson method, but requires considerably less time per iteration, and a power flow solution is obtained very rapidly. This technique is very useful in contingency analysis where numerous outages are to be simulated or a power flow solution is required for on-line control.

Example 6.12

Obtain the power flow solution by the fast decoupled method for the system of Example 6.8.

The bus admittance matrix of the system as obtained in Example 6.10 is

$$Y_{bus} = \begin{bmatrix} 20 - j50 & -10 + j20 & -10 + j30 \\ -10 + j20 & 26 - j52 & -16 + j32 \\ -10 + j30 & -16 + j32 & 26 - j62 \end{bmatrix}$$

In this system, bus 1 is the slack bus and the corresponding bus susceptance matrix for evaluation of phase angles $\Delta\delta_2$ and $\Delta\delta_3$ is

$$B' = \begin{bmatrix} -52 & 32 \\ 32 & -62 \end{bmatrix}$$

The inverse of the above matrix is

$$[B']^{-1} = \begin{bmatrix} -0.028182 & -0.014545 \\ -0.014545 & -0.023636 \end{bmatrix}$$

From (6.52) and (6.53), the expressions for real power at bus 2 and 3 and the reactive power at bus 2 are

$$\begin{aligned} P_2 &= |V_2||V_1||Y_{21}|\cos(\theta_{21} - \delta_2 + \delta_1) + |V_2^2||Y_{22}|\cos\theta_{22} \\ &\quad + |V_2||V_3||Y_{23}|\cos(\theta_{23} - \delta_2 + \delta_3) \\ P_3 &= |V_3||V_1||Y_{31}|\cos(\theta_{31} - \delta_3 + \delta_1) + |V_3||V_2||Y_{32}|\cos(\theta_{32} \\ &\quad - \delta_3 + \delta_2) + |V_3^2||Y_{33}|\cos\theta_{33} \\ Q_2 &= -|V_2||V_1||Y_{21}|\sin(\theta_{21} - \delta_2 + \delta_1) - |V_2^2||Y_{22}|\sin\theta_{22} \\ &\quad - |V_2||V_3||Y_{23}|\sin(\theta_{23} - \delta_2 + \delta_3) \end{aligned}$$

The load and generation expressed in per units are

$$S_2^{sch} = -\frac{(400 + j250)}{100} = -4.0 - j2.5 \quad \text{pu}$$

$$P_3^{sch} = \frac{200}{100} = 2.0 \quad \text{pu}$$

The slack bus voltage is $V_1 = 1.05\angle 0$ pu, and the bus 3 voltage magnitude is $|V_3| = 1.04$ pu. Starting with an initial estimate of $|V_2^{(0)}| = 1.0$, $\delta_2^{(0)} = 0.0$, and $\delta_3^{(0)} = 0.0$, the power residuals are computed from (6.63) and (6.64)

$$\begin{aligned} \Delta P_2^{(0)} &= P_2^{sch} - P_2^{(0)} = -4.0 - (-1.14) = -2.86 \\ \Delta P_3^{(0)} &= P_3^{sch} - P_3^{(0)} = 2.0 - (0.5616) = 1.4384 \\ \Delta Q_2^{(0)} &= Q_2^{sch} - Q_2^{(0)} = -2.5 - (-2.28) = -0.22 \end{aligned}$$

The fast decoupled power flow algorithm given by (6.77) becomes

$$\begin{bmatrix} \Delta\delta_2^{(0)} \\ \Delta\delta_3^{(0)} \end{bmatrix} = -\begin{bmatrix} -0.028182 & -0.014545 \\ -0.014545 & -0.023636 \end{bmatrix} \begin{bmatrix} \frac{-2.8600}{1.0} \\ \frac{1.4384}{1.04} \end{bmatrix} = \begin{bmatrix} -0.060483 \\ -0.008909 \end{bmatrix}$$

Since bus 3 is a regulated bus, the corresponding row and column of $\boldsymbol{B'}$ are eliminated and we get

$$B'' = [-52]$$

From (6.78), we have

$$\Delta|V_2| = -\left[\frac{-1}{52}\right]\left[\frac{-.22}{1.0}\right] = -0.0042308$$

The new bus voltages in the first iteration are

$$\begin{aligned} \Delta\delta_2^{(0)} &= -0.060483 & \delta_2^{(1)} &= 0 + (-0.060483) = -0.060483 \\ \Delta\delta_3^{(0)} &= -0.008989 & \delta_3^{(1)} &= 0 + (-0.008989) = -0.008989 \\ \Delta|V_2^{(0)}| &= -0.0042308 & |V_2^{(1)}| &= 1 + (-0.0042308) = 0.995769 \end{aligned}$$

The voltage phase angles are in radians. The process is continued until power residuals are within a specified accuracy. The result is tabulated in the table below.

Iter	δ_2	δ_3	$\lvert V_2\rvert$	ΔP_2	ΔP_3	ΔQ_2
1	-0.060482	-0.008909	0.995769	-2.860000	1.438400	-0.220000
2	-0.056496	-0.007952	0.965274	0.175895	-0.070951	-1.579042
3	-0.044194	-0.008690	0.965711	0.640309	-0.457039	0.021948
4	-0.044802	-0.008986	0.972985	-0.021395	0.001195	0.365249
5	-0.047665	-0.008713	0.973116	-0.153368	0.112899	0.006657
6	-0.047614	-0.008645	0.971414	0.000520	0.002610	-0.086136
7	-0.046936	-0.008702	0.971333	0.035980	-0.026190	-0.004067
8	-0.046928	-0.008720	0.971732	0.000948	-0.001411	0.020119
9	-0.047087	-0.008707	0.971762	-0.008442	0.006133	0.001558
10	-0.047094	-0.008702	0.971669	-0.000470	0.000510	-0.004688
11	-0.047057	-0.008705	0.971660	0.001971	-0.001427	-0.000500
12	-0.047054	-0.008706	0.971681	0.000170	-0.000163	0.001087
13	-0.047063	-0.008706	0.971684	-0.000458	0.000330	0.000151
14	-0.047064	-0.008706	0.971680	-0.000053	0.000048	-0.000250

Converting phase angles to degrees the final solution is $V_2 = 0.97168\angle -2.696°$ and $V_3 = 1.04\angle -0.4988°$. Using (6.52) and (6.53) as in Example 6.10, the reactive

power at bus 3 and the slack bus real and reactive powers are

$$Q_3 = 1.4617 \text{ pu}$$
$$P_1 = 2.1842 \text{ pu}$$
$$Q_1 = 1.4085 \text{ pu}$$

The fast decoupled power flow for this example has taken 14 iterations with the maximum power mismatch of 2.5×10^{-4} pu compared to the Newton-Raphson method which took only three iterations. The highest X/R ratio of the transmission lines in this example is 3. For systems with a higher X/R ratio, the fast decoupled power flow method converges in relatively fewer iterations. However, the number of iterations is a function of system size.

Finally, the line flows are calculated in the same manner as the line flow calculations in the Gauss-Seidel method described in Example 6.7, and the power flow diagram is as shown in Figure 6.13.

A program named **decouple** is developed for power flow solution by the fast decoupled method for practical power systems. This program must be preceded by the **lfybus** program. **busout** and **lineflow** programs can be used to print the load flow solution and the line flow results. The format is the same as the Gauss-Seidel method. The following is a brief description of the **decouple** program:

decouple This program finds the power flow solution by the fast decouple method and requires the **busdata** and the **linedata** files described in Section 6.9. It is designed for the direct use of load and generation in MW and Mvar, bus voltages in per unit, and angle in degrees. Loads and generation are converted to per unit quantities on the base MVA selected. A provision is made to maintain the generator reactive power of the voltage-controlled buses within their specified limits. The violation of reactive power limit may occur if the specified voltage is either too high or too low. In the 10th iteration, the vars calculated at the generator buses are examined. If a limit is reached, the voltage magnitude is adjusted in steps of 0.5 percent up to ± 5 percent to bring the var demand within the specified limits.

Example 6.13

Obtain the power flow solution for the IEEE-30 bus test system by the fast decoupled method.

Data required is the same as in Example 6.9 with the following commands

```
clear      % clears all variables from the workspace.
basemva = 100;  accuracy = 0.001;  maxiter = 20;

busdata =  [ same as in Example 6.9 ];
linedata = [ same as in Example 6.9 ];

     lfybus                     % Forms the bus admittance matrix
     decouple    % Power flow solution by fast decoupled method
     busout      % Prints the power flow solution on the screen
     lineflow % Computes and displays the line flow and losses
```

The output of **decouple** is

```
          Power Flow Solution by Fast Decoupled Method
             Maximum Power mismatch = 0.000919582
                    No. of iterations = 15

Bus Voltage  Angle    -----Load-----   --Generation-- Injected
No. Mag.     Degree    MW       Mvar     MW      Mvar     Mvar

1   1.060    0.000   0.000    0.000  260.998  -17.021    0.00
2   1.043   -5.497  21.700   12.700   40.000   48.822    0.00
3   1.022   -8.004   2.400    1.200    0.000    0.000    0.00
4   1.013   -9.662   7.600    1.600    0.000    0.000    0.00
5   1.010  -14.381  94.200   19.000    0.000   35.975    0.00
6   1.012  -11.398   0.000    0.000    0.000    0.000    0.00
7   1.003  -13.149  22.800   10.900    0.000    0.000    0.00
8   1.010  -12.115  30.000   30.000    0.000   30.828    0.00
9   1.051  -14.434   0.000    0.000    0.000    0.000    0.00
10  1.044  -16.024   5.800    2.000    0.000    0.000   19.00
11  1.082  -14.434   0.000    0.000    0.000   16.120    0.00
12  1.057  -15.303  11.200    7.500    0.000    0.000    0.00
13  1.071  -15.303   0.000    0.000    0.000   10.421    0.00
14  1.042  -16.198   6.200    1.600    0.000    0.000    0.00
15  1.038  -16.276   8.200    2.500    0.000    0.000    0.00
16  1.045  -15.881   3.500    1.800    0.000    0.000    0.00
17  1.039  -16.188   9.000    5.800    0.000    0.000    0.00
18  1.028  -16.882   3.200    0.900    0.000    0.000    0.00
19  1.025  -17.051   9.500    3.400    0.000    0.000    0.00
20  1.029  -16.852   2.200    0.700    0.000    0.000    0.00
21  1.032  -16.468  17.500   11.200    0.000    0.000    0.00
22  1.033  -16.454   0.000    0.000    0.000    0.000    0.00
23  1.027  -16.661   3.200    1.600    0.000    0.000    0.00
24  1.022  -16.829   8.700    6.700    0.000    0.000    4.30
25  1.019  -16.423   0.000    0.000    0.000    0.000    0.00
26  1.001  -16.840   3.500    2.300    0.000    0.000    0.00
```

```
27  1.026  -15.912    0.000    0.000    0.000    0.000    0.00
28  1.011  -12.057    0.000    0.000    0.000    0.000    0.00
29  1.006  -17.136    2.400    0.900    0.000    0.000    0.00
30  0.995  -18.014   10.600    1.900    0.000    0.000    0.00

Total                283.400  126.200  300.998  125.145   23.30
```

The output of the **lineflow** is the same as the line flow output of Example 6.9 with the power mismatch as dictated by the fast decoupled method.

PROBLEMS

6.1. A power system network is shown in Figure 6.17. The generators at buses 1 and 2 are represented by their equivalent current sources with their reactances in per unit on a 100-MVA base. The lines are represented by π model where series reactances and shunt reactances are also expressed in per unit on a 100 MVA base. The loads at buses 3 and 4 are expressed in MW and Mvar.

(a) Assuming a voltage magnitude of 1.0 per unit at buses 3 and 4, convert the loads to per unit impedances. Convert network impedances to admittances and obtain the bus admittance matrix by inspection.

(b) Use the function **Y = ybus(zdata)** to obtain the bus admittance matrix. The function argument **zdata** is a matrix containing the line bus numbers, resistance and reactance. (See Example 6.1.)

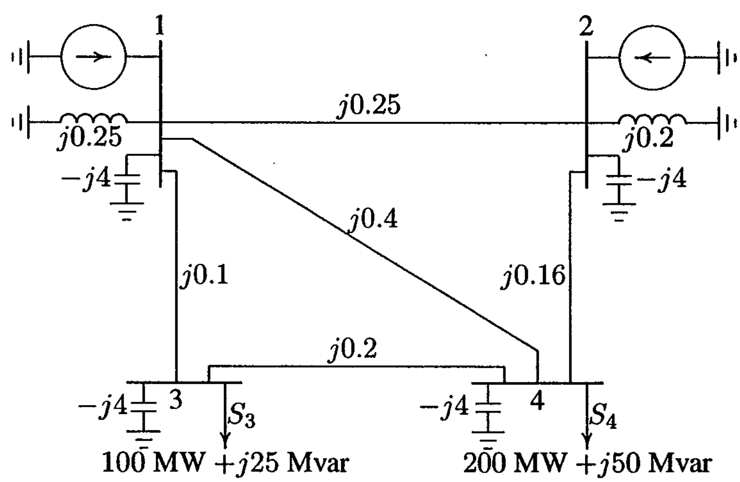

FIGURE 6.17
One-line diagram for Problem 6.1.

6.2. A power system network is shown in Figure 6.18. The values marked are impedances in per unit on a base of 100 MVA. The currents entering buses 1 and 2 are

$$I_1 = 1.38 - j2.72 \text{ pu}$$
$$I_2 = 0.69 - j1.36 \text{ pu}$$

(a) Determine the bus admittance matrix by inspection.
(b) Use the function **Y = ybus(zdata)** to obtain the bus admittance matrix. The function argument **zdata** is a matrix containing the line bus numbers, resistance and reactance. (See Example 6.1.) Write the necessary *MATLAB* commands to obtain the bus voltages.

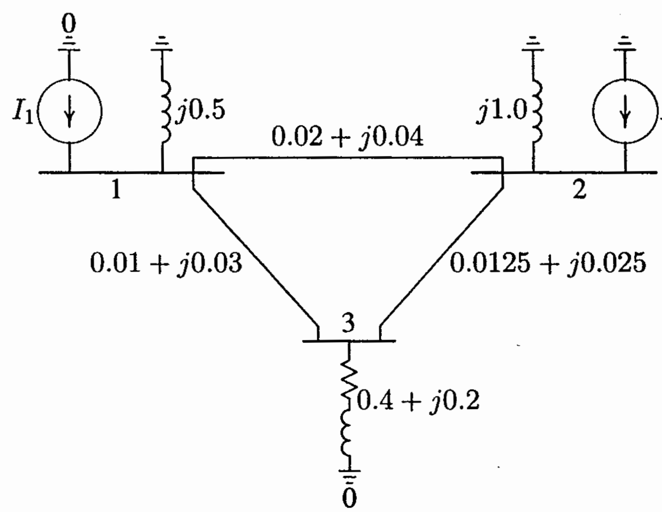

FIGURE 6.18
One-line diagram for Problem 6.2.

6.3. Use Gauss-Seidel method to find the solution of the following equations

$$x_1 + x_1 x_2 = 10$$
$$x_1 + x_2 = 6$$

with the following initial estimates
(a) $x_1^{(0)} = 1$ and $x_2^{(0)} = 1$
(b) $x_1^{(0)} = 1$ and $x_2^{(0)} = 2$
Continue the iterations until $|\Delta x_1^{(k)}|$ and $|\Delta x_2^{(k)}|$ are less than 0.001.

6.4. A fourth-order polynomial equation is given by

$$x^4 - 21x^3 + 147x^2 - 379x + 252 = 0$$

(a) Use Newton-Raphson method and hand calculations to find one of the roots of the polynomial equation. Start with the initial estimate of $x^{(0)} = 0$ and continue until $|\Delta x^{(k)}| < 0.001$.
(b) Write a *MATLAB* program to find the roots of the above polynomial by Newton-Raphson method. The program should prompt the user to input the initial estimate. Run using the initial estimates of 0, 3, 6, 10.
(c) Check your answers using the *MATLAB* function **r = roots(A)**, where **A** is a row vector containing the polynomial coefficients in descending powers.

6.5. Use Newton-Raphson method and hand calculation to find the solution of the following equations:

$$x_1^2 - 2x_1 - x_2 = 3$$
$$x_1^2 + x_2^2 = 41$$

(a) Start with the initial estimates of $x_1^{(0)} = 2$, $x_2^{(0)} = 3$. Perform three iterations.
(b) Write a *MATLAB* program to find one of the solutions of the above equations by Newton-Raphson method. The program should prompt the user to input the initial estimates. Run the program with the above initial estimates.

6.6. In the power system network shown in Figure 6.19, bus 1 is a slack bus with $V_1 = 1.0\angle 0°$ per unit and bus 2 is a load bus with $S_2 = 280$ MW + $j60$ Mvar. The line impedance on a base of 100 MVA is $Z = 0.02 + j0.04$ per unit.
(a) Using Gauss-Seidel method, determine V_2. Use an initial estimate of $V_2^{(0)} = 1.0 + j0.0$ and perform four iterations.
(b) If after several iterations voltage at bus 2 converges to $V_2 = 0.90 - j0.10$, determine S_1 and the real and reactive power loss in the line.

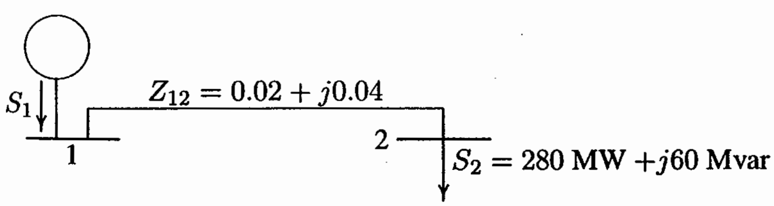

FIGURE 6.19
One-line diagram for Problem 6.6.

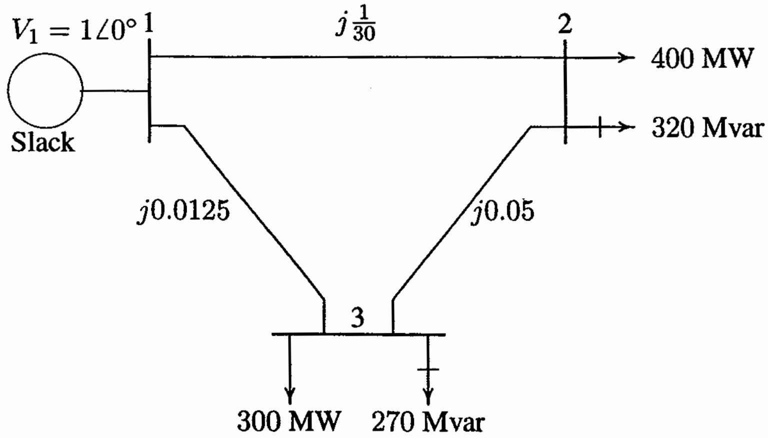

FIGURE 6.20
One-line diagram for Problem 6.7.

6.7. Figure 6.20 shows the one-line diagram of a simple three-bus power system with generation at bus 1. The voltage at bus 1 is $V_1 = 1.0\angle 0°$ per unit. The scheduled loads on buses 2 and 3 are marked on the diagram. Line impedances are marked in per unit on a 100-MVA base. For the purpose of hand calculations, line resistances and line charging susceptances are neglected.

(a) Using Gauss-Seidel method and initial estimates of $V_2^{(0)} = 1.0 + j0$ and $V_3^{(0)} = 1.0 + j0$, determine V_2 and V_3. Perform two iterations.

(b) If after several iterations the bus voltages converge to

$$V_2 = 0.90 - j0.10 \text{ pu}$$
$$V_3 = 0.95 - j0.05 \text{ pu}$$

determine the line flows and line losses and the slack bus real and reactive power. Construct a power flow diagram and show the direction of the line flows.

(c) Check the power flow solution using the **lfgauss** and other required programs. (Refer to Example 6.9.) Use a power accuracy of 0.00001 and an acceleration factor of 1.0.

6.8. Figure 6.21 shows the one-line diagram of a simple three-bus power system with generation at buses 1 and 3. The voltage at bus 1 is $V_1 = 1.025\angle 0°$ per unit. Voltage magnitude at bus 3 is fixed at 1.03 pu with a real power generation of 300 MW. A load consisting of 400 MW and 200 Mvar is taken from bus 2. Line impedances are marked in per unit on a 100-MVA base. For the

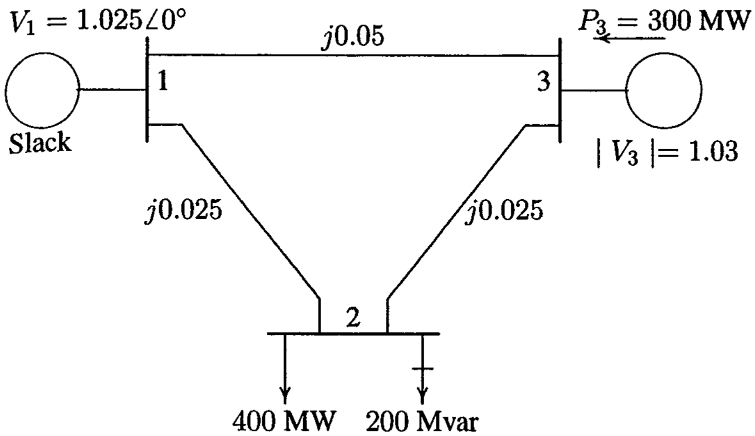

FIGURE 6.21
One-line diagram for Problem 6.8.

purpose of hand calculations, line resistances and line charging susceptances are neglected.

(a) Using Gauss-Seidel method and initial estimates of $V_2^{(0)} = 1.0 + j0$ and $V_3^{(0)} = 1.03 + j0$ and keeping $|V_3| = 1.03$ pu, determine the phasor values of V_2 and V_3 . Perform two iterations.

(b) If after several iterations the bus voltages converge to

$$V_2 = 1.001243\angle -2.1° = 1.000571 - j0.0366898 \text{ pu}$$
$$V_3 = 1.03\angle 1.36851° = 1.029706 + j0.0246 \text{ pu}$$

determine the line flows and line losses and the slack bus real and reactive power. Construct a power flow diagram and show the direction of the line flows.

(c) Check the power flow solution using the **lfgauss** and other required programs. (Refer to Example 6.9.)

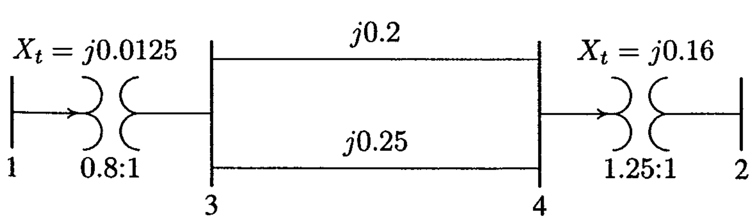

FIGURE 6.22
One-line diagram for Problem 6.9.

6.9. The one-line diagram of a four-bus power system is as shown in Figure 6.22. Reactances are given in per unit on a common MVA base. Transformers T_1 and T_2 have tap settings of 0.8:1, and 1.25:1 respectively. Obtain the bus admittance matrix.

6.10. In the two-bus system shown in Figure 6.23, bus 1 is a slack bus with $V_1 = 1.0\angle 0°$ pu. A load of 150 MW and 50 Mvar is taken from bus 2. The line admittance is $y_{12} = 10\angle -73.74°$ pu on a base of 100 MVA. The expression for real and reactive power at bus 2 is given by

$$
\begin{aligned}
P_2 &= 10|V_2||V_1|\cos(106.26° - \delta_2 + \delta_1) + 10|V_2|^2\cos(-73.74°)\\
Q_2 &= -10|V_2||V_1|\sin(106.26° - \delta_2 + \delta_1) - 10|V_2|^2\sin(-73.74°)
\end{aligned}
$$

Using Newton-Raphson method, obtain the voltage magnitude and phase angle of bus 2. Start with an initial estimate of $|V_2|^{(0)} = 1.0$ pu and $\delta_2^{(0)} = 0°$. Perform two iterations.

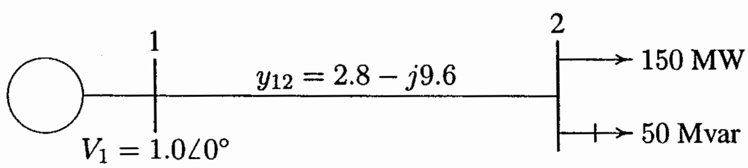

FIGURE 6.23
One-line diagram for Problem 6.10.

6.11. In the two-bus system shown in Figure 6.24, bus 1 is a slack bus with $V_1 = 1.0\angle 0°$ pu. A load of 100 MW and 50 Mvar is taken from bus 2. The line impedance is $z_{12} = 0.12 + j0.16$ pu on a base of 100 MVA. Using Newton-Raphson method, obtain the voltage magnitude and phase angle of bus 2. Start with an initial estimate of $|V_2|^{(0)} = 1.0$ pu and $\delta_2^{(0)} = 0°$. Perform two iterations.

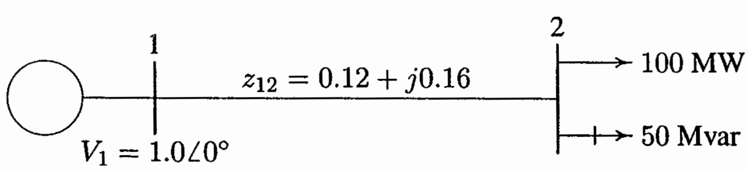

FIGURE 6.24
One-line diagram for Problem 6.11.

6.12. Figure 6.25 shows the one-line diagram of a simple three-bus power system with generation at buses 1 and 2. The voltage at bus 1 is $V = 1.0\angle 0^\circ$ per unit. Voltage magnitude at bus 2 is fixed at 1.05 pu with a real power generation of 400 MW. A load consisting of 500 MW and 400 Mvar is taken from bus 3. Line admittances are marked in per unit on a 100 MVA base. For the purpose of hand calculations, line resistances and line charging susceptances are neglected.

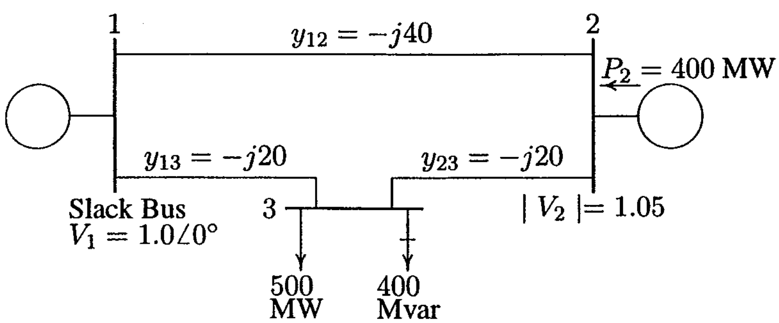

FIGURE 6.25
One-line diagram for Problem 6.12

(a) Show that the expression for the real power at bus 2 and real and reactive power at bus 3 are

$$P_2 = 40|V_2||V_1|\cos(90^\circ - \delta_2 + \delta_1) + 20|V_2||V_3|\cos(90^\circ - \delta_2 + \delta_3)$$
$$P_3 = 20|V_3||V_1|\cos(90^\circ - \delta_3 + \delta_1) + 20|V_3||V_2|\cos(90^\circ - \delta_3 + \delta_2)$$
$$Q_3 = -20|V_3||V_1|\sin(90^\circ - \delta_3 + \delta_1) - 20|V_3||V_2|\sin(90^\circ - \delta_3 + \delta_2) + 40|V_3|^2$$

(b) Using Newton-Raphson method, start with the initial estimates of $V_2^{(0)} = 1.0 + j0$ and $V_3^{(0)} = 1.0 + j0$, and keeping $|V_2| = 1.05$ pu, determine the phasor values of V_2 and V_3. Perform two iterations.
(c) Check the power flow solution for Problem 6.12 using **lfnewton** and other required programs. Assume the regulated bus (bus # 2) reactive power limits are between 0 and 600 Mvar.

6.13. For Problem 6.12:
(a) Obtain the power flow solution using the fast decoupled algorithm. Perform two iterations.
(b) Check the power flow solution for Problem 6.12 using **decouple** and other required programs. Assume the regulated bus (bus # 2) reactive power limits are between 0 and 600 Mvar.

6.14. The 26-bus power system network of an electric utility company is shown in Figure 6.26 (page 256). Obtain the power flow solution by the following

methods:
(a) Gauss-Seidel power flow (see Example 6.9).
(b) Newton-Raphson power flow (see Example 6.11).
(c) Fast decoupled power flow (see Example 6.13).

The load data is as follows.

LOAD DATA					
Bus	Load		Bus	Load	
No.	MW	Mvar	No.	MW	Mvar
1	51.0	41.0	14	24.0	12.0
2	22.0	15.0	15	70.0	31.0
3	64.0	50.0	16	55.0	27.0
4	25.0	10.0	17	78.0	38.0
5	50.0	30.0	18	153.0	67.0
6	76.0	29.0	19	75.0	15.0
7	0.0	0.0	20	48.0	27.0
8	0.0	0.0	21	46.0	23.0
9	89.0	50.0	22	45.0	22.0
10	0.0	0.0	23	25.0	12.0
11	25.0	15.0	24	54.0	27.0
12	89.0	48.0	25	28.0	13.0
13	31.0	15.0	26	40.0	20.0

Voltage magnitude, generation schedule, and the reactive power limits for the regulated buses are tabulated below. Bus 1, whose voltage is specified as $V_1 = 1.025\angle 0°$, is taken as the slack bus.

GENERATION DATA				
Bus	Voltage	Generation	Mvar Limits	
No.	Mag.	MW	Min.	Max.
1	1.025			
2	1.020	79.0	40.0	250.0
3	1.025	20.0	40.0	150.0
4	1.050	100.0	40.0	80.0
5	1.045	300.0	40.0	160.0
26	1.015	60.0	15.0	50.0

The Mvar of the shunt capacitors installed at substations and the transformer tap settings are given below.

SHUNT CAPACITORS	
Bus No.	Mvar
1	4.0
4	2.0
5	5.0
6	2.0
11	1.5
12	2.0
15	0.5
19	5.0

TRANSFORMER TAP	
Designation	Tap Setting
2 – 3	0.960
2 – 13	0.960
3 – 13	1.017
4 – 8	1.050
4 – 12	1.050
6 – 19	0.950
7 – 9	0.950

The line and transformer data containing the series resistance and reactance in per unit and one-half the total capacitance in per unit susceptance on a 100-MVA base are tabulated below.

LINE AND TRANSFORMER DATA									
Bus No.	Bus No.	R, pu	X, pu	$\frac{1}{2}B$, pu	Bus No.	Bus No.	R, pu	X, pu	$\frac{1}{2}B$, pu
1	2	0.0005	0.0048	0.0300	10	22	0.0069	0.0298	0.005
1	18	0.0013	0.0110	0.0600	11	25	0.0960	0.2700	0.010
2	3	0.0014	0.0513	0.0500	11	26	0.0165	0.0970	0.004
2	7	0.0103	0.0586	0.0180	12	14	0.0327	0.0802	0.000
2	8	0.0074	0.0321	0.0390	12	15	0.0180	0.0598	0.000
2	13	0.0035	0.0967	0.0250	13	14	0.0046	0.0271	0.001
2	26	0.0323	0.1967	0.0000	13	15	0.0116	0.0610	0.000
3	13	0.0007	0.0054	0.0005	13	16	0.0179	0.0888	0.001
4	8	0.0008	0.0240	0.0001	14	15	0.0069	0.0382	0.000
4	12	0.0016	0.0207	0.0150	15	16	0.0209	0.0512	0.000
5	6	0.0069	0.0300	0.0990	16	17	0.0990	0.0600	0.000
6	7	0.0053	0.0306	0.0010	16	20	0.0239	0.0585	0.000
6	11	0.0097	0.0570	0.0001	17	18	0.0032	0.0600	0.038
6	18	0.0037	0.0222	0.0012	17	21	0.2290	0.4450	0.000
6	19	0.0035	0.0660	0.0450	19	23	0.0300	0.1310	0.000
6	21	0.0050	0.0900	0.0226	19	24	0.0300	0.1250	0.002
7	8	0.0012	0.0069	0.0001	19	25	0.1190	0.2249	0.004
7	9	0.0009	0.0429	0.0250	20	21	0.0657	0.1570	0.000
8	12	0.0020	0.0180	0.0200	20	22	0.0150	0.0366	0.000
9	10	0.0010	0.0493	0.0010	21	24	0.0476	0.1510	0.000
10	12	0.0024	0.0132	0.0100	22	23	0.0290	0.0990	0.000
10	19	0.0547	0.2360	0.0000	22	24	0.0310	0.0880	0.000
10	20	0.0066	0.0160	0.0010	23	25	0.0987	0.1168	0.000

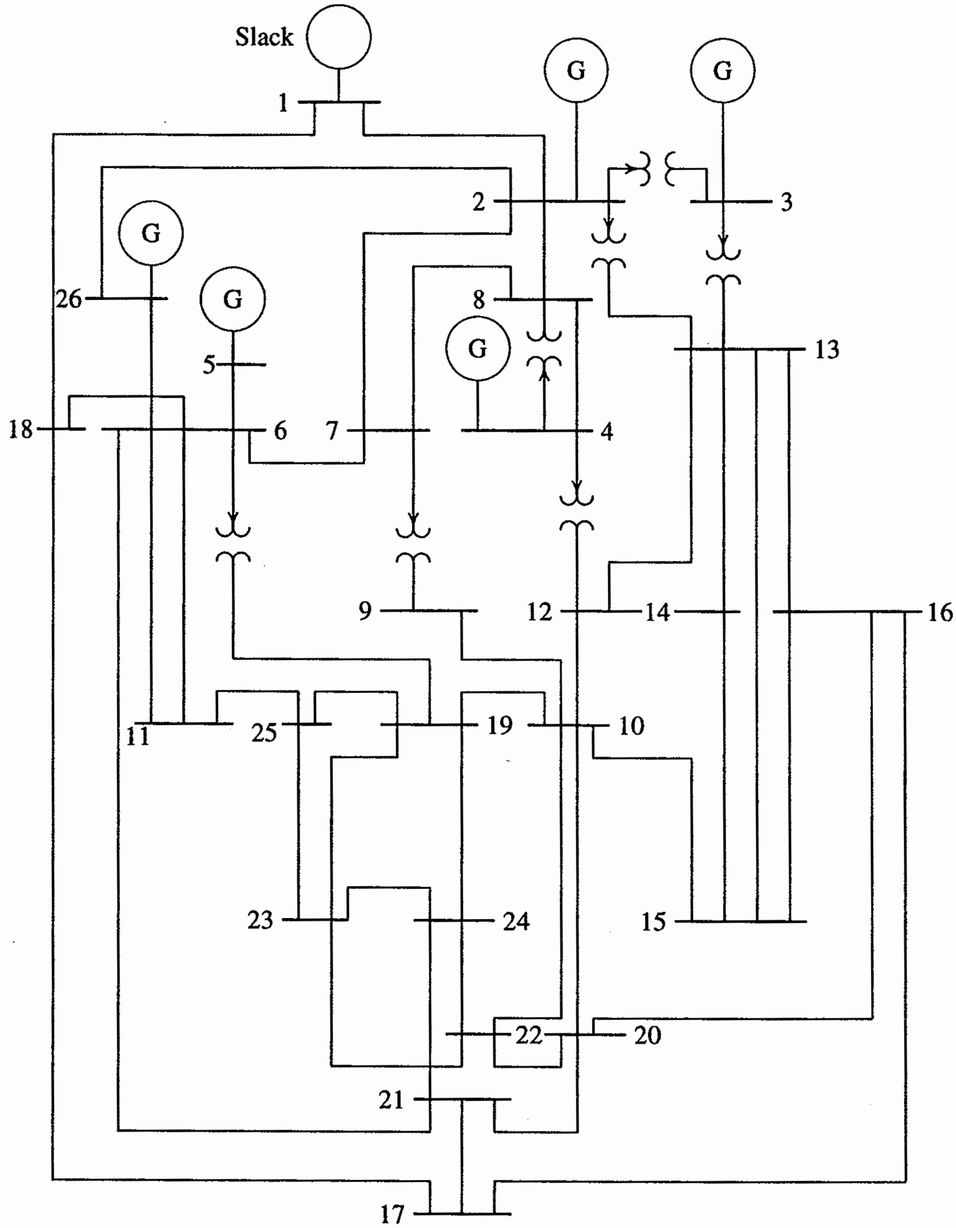

FIGURE 6.26
One-line diagram for Problem 6.14.